

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

PABLO ILHA VAZ

PROJETO DE DIPLOMAÇÃO

SÍNTESE DIGITAL DIRETA

Porto Alegre
2012

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

SÍNTESE DIGITAL DIRETA

Projeto de Diplomação apresentado ao Departamento de Engenharia Elétrica da Universidade Federal do Rio Grande do Sul, como parte dos requisitos para Graduação em Engenharia Elétrica.

ORIENTADOR: Prof. MSc. Tiaraju Vasconcellos
Wagner

Porto Alegre
2012

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

PABLO ILHA VAZ

SÍNTESE DIGITAL DIRETA

TUTORIAL E ESTUDO DE CARACTERÍSTICAS NÃO LINEARES

Este projeto foi julgado adequado para fazer jus aos créditos da Disciplina de “Projeto de Diplomação”, do Departamento de Engenharia Elétrica e aprovado em sua forma final pelo Orientador e pela Banca Examinadora.

Orientador: _____
Prof. MSc. Tiaraju Vasconcellos Wagner, UFRGS
Mestre pela UFRGS – Porto Alegre, Brasil.

Banca Examinadora:

Prof. MSc. Tiaraju Vasconcellos Wagner, UFRGS
Mestre pela UFRGS – Porto Alegre, Brasil.

Prof. Dr. Hamilton Duarte Klimach, UFRGS
Doutor pela UFSC – Florianópolis, Brasil.

Eng. André Luiz Pranke, OI (MSc, PmP)
Mestre pela Hochschule – Darmstadt, Alemanha.

Porto Alegre, março de 2012.

DEDICATÓRIA

Dedico este trabalho aos meus pais que contribuíram com paciência, carinho e compreensão, que sempre me deram força em todos os dias de prova, me acompanharam no preparo, arcaram com as despesas de psiquiatra, aguentaram meus frequentes momentos de mau humor e agora estão comemorando comigo o final desta etapa.

AGRADECIMENTOS

À minha família a qual sempre conseguiu me proporcionar o melhor ambiente de estudos possível, que me ensinou que o conhecimento é uma das maiores riquezas que possuímos e o único bem que não pode ser tirado do ser humano.

Aos colegas Maurício Grando, Sílvio Asmann, Clarissa Bastos, Roberto Volkmann e Leonel Guimarães pela amizade, ajuda e auxílio muitas vezes psicológico principalmente nas tarefas finais do curso.

Ao meu orientador Prof. MSc. Tiaraju Vasconcellos Wagner, pela confiança depositada como orientador desse projeto, pela orientação objetiva e segura, e pela amizade. Suas orientações em muito contribuíram para o meu desenvolvimento acadêmico.

À minha querida namorada Gabriela Zuchi por sempre me incentivar e acreditar em mim muitas vezes mais do que eu próprio e que nunca me deixou esquecer que tudo dependia do quanto eu realmente quisesse.

“Nosso destino está de acordo com o trabalho que realizamos e a nossa felicidade com o prazer que temos ao trabalhar”

Pablo Ilha Vaz

RESUMO

Este trabalho tem como propósito fornecer subsídios, como um manual, para que estudantes e profissionais da área de engenharia elétrica executem projetos e pesquisas em que seja utilizado o conceito da Síntese Digital Direta (*DDS*).

São abordados todos os tópicos pertinentes do *DDS*, considerando desde a explicação básica dos conceitos, funcionamento e análise de blocos funcionais até considerações sobre comportamentos limites, não linearidades e estudo da geração de variáveis espúrias, analisando a qualidade senoidal sintetizada e a sua relação com os dados gravados na memória bem como as características e qualidade do conversor digital-analógico.

Adicionalmente, é feita uma profunda revisão bibliográfica sobre o assunto, organizando e apresentando os tópicos mais relevantes para que análises de projeto e implementações práticas possam ser realizadas, fundamentadas neste material teórico realmente embasado.

Palavras-chave: *DDS*, Manual, Síntese Digital Direta, Espúrios, Não linearidade.

ABSTRACT

This work aims to provide information as a guide for students and professionals in the field of electrical engineering can execute projects and perform research that use the concept of Direct Digital Synthesis (*DDS*).

Are discussed all relevant and important topics about *DDS*, considering since the explanation of basics, operation and analysis of functional blocks to considerations limits on behavior, nonlinearities and study the generation of spurious variables, analyzing the quality sine synthesized and its relation to the stored data in memory and the both characteristics and quality of the digital-analog converter.

Additionally, there is a deep review of the literature about theses subjects, organizing and presenting the most relevant topics in order that design reviews and practical implementations can be performed based on this consistent theoretical material.

Keywords: *DDS*, Manual, Direct Digital Synthesis, Spurious, nonlinearity.

SUMÁRIO

INTRODUÇÃO	12
1. INTRODUÇÃO AOS SINTETIZADORES DE FREQUÊNCIA	17
1.1. Sintetizadores de frequência.....	17
1.2. Aplicações.....	18
2. CONCEITOS BÁSICOS SOBRE DDS.....	20
2.1. O que é <i>DDS</i>	20
2.2. O sistema <i>DDS</i>	20
3. ANÁLISE DA ARQUITETURA EM BLOCOS FUNCIONAIS	26
3.1. Acumulador de Fase.....	26
3.2. Tabela de conversão <i>LUT</i> / Conversor fase para amplitude	33
3.3. Conversão digital-analógico	38
3.4. Considerações técnicas adicionais.....	39
4. IMPLEMENTAÇÃO PRÁTICA - SIMULAÇÕES.....	44
4.1. Saída digital de fase.....	44
4.2. Saída do conversor digital-analógico	46
4.3. Saída do conversor digital-analógico com filtro passa-baixa	47
4.4. Efeito do ruído de quantização do sinal do conversor digital-analógico.....	48
4.5. Espectro do sinal de saída do conversor digital-analógico	48
5. TRABALHOS FUTUROS.....	50
5.1. Truncamento de fase / bits.....	50
5.1.1. Magnitude espúria do truncamento de fase	56
5.2. Simetria do Quarto de Onda	61
5.3. Teorema da Amostragem.....	63
5.4. Quantização.....	65
5.4.1. Efeitos de sobre-amostragem em performance espúria.....	67
5.5. Jitter.....	68
CONCLUSÕES.....	70
REFERÊNCIAS	71
APÊNDICE	73
Matlab® - Código 1	73
Matlab® - Código 2	74
Matlab® - Código 3	75
Matlab® - Código 4	76
Matlab® - Código 5	77
Matlab® - Código 6	78

LISTA DE ILUSTRAÇÕES

Figura 1: Esquema básico de um sistema de Síntese Digital Direta.	20
Figura 2: Magnitude da senoide e representações de fase.....	22
Figura 3: Amplitude da senoide armazenada na <i>LUT</i>	25
Figura 4: Saída digital de fase.	25
Figura 5: Saída do <i>DAC</i>	26
Figura 6: Célula mínima de um acumulador de fase.	27
Figura 7: Círculo de fases.	27
Figura 8: Interpretação dos valores do círculo de fase.	29
Figura 9: Computação do registrador de fase.....	30
Figura 10: Configuração para $M=1$ e $M=2$	34
Figura 11: Relação entre o incremento de fase e a taxa de variação da fase.....	37
Figura 12: Memória ideal (infinita).	37
Figura 13: Resolução do <i>DAC</i>	39
Figura 14: Esquema geral resumido da síntese	39
Figura 15: Saída digital de fase caso 1	45
Figura 16: Saída digital de fase caso 2	45
Figura 17: Saída do conversor digital-analógico caso 1.....	46
Figura 18: Saída do conversor digital-analógico caso 2.....	47
Figura 19: Saída do conversor digital-analógico com filtro de ordem 2.....	47
Figura 20: Efeito do ruído de quantização do sinal de saída do conversor digital-analógico ..	48
Figura 21: Espectro do sinal de saída do conversor digital-analógico ..	49
Figura 22: Palavra de ajuste de 32 bits.	50
Figura 23: Palavra de ajuste de 32 bits com truncamento de fase.....	51
Figura 24: Erro de truncamento de fase e roda de fase	52
Figura 25: Acumulador de fase não truncado.....	54
Figura 26: Saída do conversor digital-analógico não truncada	54
Figura 27: Saída do acumulador truncada.	55
Figura 28: Saída do conversor digital-analógico truncada.	55
Figura 29: Palavra de ajuste padrão correspondente a máxima magnitude de espúrios.....	58
Figura 30: Palavra de ajuste padrão que não resulta em erro espúrios.....	59
Figura 31: Relação do <i>Jitter</i> com o incremento de fase.	61
Figura 32: Simetria do quarto de onda.	62
Figura 33: Esquema <i>DDS</i> com compressão de quarto de onda.....	62
Figura 34: Espectro de amostras do sinal de saída.	64
Figura 35: Curva característica da quantização	66
Figura 36: Efeito de sobre-amostragem em <i>SQR</i>	68
Figura 37: Sinal com instabilidades devidas ao <i>Jitter</i>	69

LISTA DE ABREVIATURAS

DDS: *Direct Digital Synthesis* – “Síntese Digital Direta”

VHF: Very High Frequency – “Frequência muito alta”

UHF: Ultra High Frequency – “Frequência ultra alta”

CDMA: Code Division Multiple Access

FH-CDMA: Frequency Hopping – Code Division Multiple Access

LUT: Look Up Table – “Forma de onda implementada em dados de memória”

ROM: Read Only Memory

MSB: Most Significant Bit - “Bit mais significante”

LSB: Last Significant Bit - “Bit menos significante”

DAC: Digital Analogic Converter – “Conversor digital-analógico”

PLL: Phase Locked Loop

SQR: Ratio of Signal Power to Quantization Noise Power – “Potência de ruído de quantização”

Jitter: “Ruído de oscilação no domínio do tempo”

VLSI: Very Large Scale Integration

SNR: Signal Noise Ratio – “Relação Sinal Ruído”

RMS: Root Mean Square – “Valor quadrado médio”

MDC: Máximo Divisor Comum

ISM: Industrial, Scientific, Medical

PLL: Phase Locked Loop

INTRODUÇÃO

A síntese digital direta foi inicialmente proposta por J. Tierney em 1971. É um método de produção digital de formas de ondas. Com o desenvolvimento da tecnologia *VLSI* e os fortes requisitos dos sistemas de comunicações modernos os sintetizadores digitais tem sido amplamente utilizados desde a década de 80. Essa é a mais recente técnica desenvolvida para síntese de frequência.

Um método de síntese que aparece como concorrente do *DDS* é o Phase Locked Loop *PLL*. Essa técnica, de um modo geral, compara a saída com um sinal de referência, ou seja, depende de uma realimentação. Por tal motivo não permite uma troca tão rápida de frequência, além de possuir um ruído de fase mais significativo.

A técnica de síntese de frequência conhecida por *PLL* é a mais conhecida entre todas. A estrutura *PLL* possui um ramo de realimentação que possibilita a correção e a convergência do sinal gerado para o valor de frequência desejado. A estrutura *PLL*, além de atuar como uma técnica de síntese de frequência, pode ser usada para outras aplicações como modulador e demodulador de FM, atenuador da modulação parasita de fase (jitter) e regenerador de portadora.

Um dos blocos analisados nessa técnica é o oscilador controlado por tensão (*VCO*), responsável por gerar um sinal de frequência proporcional à tensão aplicada em sua entrada. Pode-se encontrar vários modelos de *VCO*'s e os mais conhecidos são os osciladores a cristal, os osciladores com circuito ressonante obtidos com elementos reativos, cabo coaxial ou cavidade ressonante e os multivibradores. Os quatro primeiros geram sinais analógicos e o último gera um sinal digital.

Os osciladores a cristal, têm como principal característica a geração do sinal a partir de algum harmônico de um cristal de frequência fundamental inferior. O segundo, terceiro e quarto tipos se diferenciam no circuito de sintonia. Podem ser controlados por um varicap, diodo capaz de variar a sua capacitância de junção em função da tensão reversa aplicada. O diodo varicap modifica as características do circuito ressonante resultante e faz a frequência do sinal de saída variar de acordo com a tensão aplicada em sua entrada.

Um próximo bloco que possui função importante é o filtro de malha, que recebe a maior atenção em um projeto de sintetizador de frequência. Isto acontece porque o filtro de malha é o circuito que tem maior influência sobre o comportamento do sinal de saída. A sua principal função será filtrar o sinal vindo do comparador de fase, permitindo que somente a componente resultante da diferença de frequência entre o sinal de entrada excite o VCO. Por esta razão, o filtro de malha é do tipo passa-baixas. Devido ao fato do VCO ser um componente de excitação crítica o filtro de malha deve ser projetado para que os ruídos gerados internamente ou externamente ao sistema não prejudiquem as características do sinal sintetizado, principalmente não contribua para aumentar o seu ruído de fase. O filtro de malha mais simples de ser implementado é composto de um resistor e um capacitor e está sujeito a todas as limitações dos filtros analógicos.

Duas importantes características determinadas pelo projeto do filtro de malha são a velocidade de chaveamento, fator crítico na detecção de sinais modulados em frequência e a rejeição ao ruído, fator crítico em projetos de sintetizadores de frequência. Existe uma relação de compromisso entre estas duas características. Quando o filtro de malha for projetado para uma resposta rápida às mudanças em sua entrada, com uma frequência de corte elevada, mas ainda menor que a frequência de comparação e próxima a esta, este filtro apresentará pequena rejeição às interferências e ruídos internos e externos. Se estiver projetado para grande

rejeição às interferências e ruídos internos e externos, o mesmo tenderá a apresentar uma resposta lenta às mudanças em sua entrada.

Como citado, os sistemas do tipo PLL obedecem os princípios de um sistema de controle realimentado. Um sistema de controle é um circuito de regulação automática onde a variável a ser controlada pode ser temperatura, velocidade, pressão, frequência, etc. Existem dois tipos básicos: o sistema de malha aberta e o de malha fechada. Um sistema de malha aberta possui um único ramo, chamado ramo direto, representado em um único bloco. Circuitos que trabalham em malha aberta não possuem ligação que possibilite a entrada do circuito acessar uma amostra do sinal de saída.

Devido ao fato de a entrada não ter acesso as informações de saída, o sistema de malha aberta não é capaz de modificar a saída de forma a seguir alterações nas condições instantâneas de operação e a saída não tem efeito sobre a ação de controle. Os circuitos que trabalham em malha aberta normalmente necessitam de calibração para trabalharem.

A desvantagem dos sistemas em malha aberta é que se existirem modificações no sistema como desgaste físico, mudanças do comportamento devido a temperatura, inserção de distúrbios internos ou externos após a calibração do mesmo, o sistema não será capaz de corrigir essas eventualidades. [2]

A grande vantagem de um sistema que utiliza a síntese digital direta em relação a outro, feito com técnicas analógicas está na precisão da frequência sintetizada no gerador estar diretamente ligada a precisão do cristal oscilador e ainda na resolução da frequência selecionada, que dependendo da quantidade de bits de um de seus parâmetro de configuração pode chegar a micro-Hertz [8].

Atualmente um sistema *DDS* é uma boa alternativa em comparação ao sintetizador analógico tradicional, devido ao baixo custo e salto extremamente rápido entre duas frequências distintas (salto com fase contínua sem overshoots ou anomalias de tempo de

acomodação típica de implementações analógicas). Considerações que atualmente se fazem presentes em tecnologias de comunicação celular e dispositivos de comunicação sem fio.

O presente trabalho estuda esse método de produção digital de formas de onda e se organiza em quatro capítulos seguindo uma ordem lógica de apresentação. A cada capítulo é descido um nível de abstração, que vai da apresentação básica, considerando um público alvo leigo, até o último capítulo que considera um público alvo estudantes de engenharia elétrica.

No primeiro capítulo é feita a contextualização e abordado o conceito básico sobre sintetizadores de frequência expondo algumas aplicações e motivação para execução do trabalho.

No segundo capítulo inicia-se a explicação mais detalhada dos blocos operacionais ainda sobre uma visão funcional.

No terceiro capítulo é feita a formalização matemática das equações de configuração de frequência, palavra de ajuste e suas relações com as saídas desejadas ainda é explorada a disponibilidade mais comumente encontrada no mercado em relação ao tamanho de palavras explorado alguns limites de operação do método.

No quarto capítulo são realizadas simulações com o recurso do software *Matlab*® a fim de que se consiga escolher dentro das arquiteturas disponíveis no mercado, uma que seja pelo menos suficiente à aplicação de interesse e para que seja possível comparar resultados práticos com teóricos.

E no último capítulo são abordados os aspectos mais técnicos em relação a qualidade da onda sintetizada, propagação de variáveis espúrias, não linearidades e truncamento de bits.

Pelo fato de ter executado uma revisão bibliográfica em diversos idiomas e também pelo fato de se tratar de um método de produção (o que sugere se fazer análises dentro de um determinado contexto), foi inevitável encontrar algumas dificuldades ao relacionar símbolos e variáveis as suas reais funções.

Como considerei de suma importância relevar, também, esses aspectos ao longo de um trabalho desse porte, utilizei minha experiência ao longo do curso para fornecer algumas informações extras ora sobre a origem do nome de algumas variáveis ora sobre a sugestão de utilização de outro para desambiguação. Acredito que essas informações adicionais, caso sejam realizadas pesquisas em outras literaturas, sejam muito úteis.

1. Introdução aos sintetizadores de frequência

1.1. Sintetizadores de frequência

Sintetizadores de frequência são circuitos capazes de gerar um ou mais sinais a partir de uma ou várias fontes de referência. A saída de um sintetizador é caracterizada por sua frequência de operação, resolução e, no caso das senoides, sua distorção harmônica. Diversas são as formas de se implementar os sintetizadores utilizando recursos digitais, analógicos, métodos diretos e indiretos e uma combinação de vários destes [19].

Na sua forma mais primitiva a síntese de frequência era realizada por um oscilador controlado a cristal, com um banco de cristais chaveados manualmente. A exatidão e estabilidade em frequência desse tipo de dispositivo é determinada pela exatidão e estabilidade do cristal e, numa extensão menor, pelo próprio circuito. Pelo fato de que novos desenvolvimentos no campo das comunicações exigiram requisitos de exatidão e estabilidade por várias ordens de grandeza superior as anteriores, evoluiu-se então para outra família de técnicas, agrupadas sob o nome de síntese coerente, a qual produz muitas frequências a partir de uma fonte de referência com mais exatidão e estabilidade que as anteriores [11].

Por fim, com o advento da integração em média e larga escala, uma nova família de técnicas de síntese, denominada síntese digital direta, tornou-se prática. Nessa as amostras da senoide de saída são geradas digitalmente, convertidas para a forma analógica por um conversor digital-analógico (*DAC*) e então filtradas para produzir a saída senoidal desejada.

1.2. Aplicações

Provavelmente o maior impulso para o desenvolvimento de técnicas de síntese de frequência e conseqüentemente de seus sintetizadores, tem sido a demanda para capacitar a existência de vários canais de comunicação dentro de uma faixa de frequência limitada.

Foi possível com os sintetizadores reduzir o espaçamento entre canais, eliminando-se a banda de proteção prevista para evitar interferências entre bandas adjacentes e acomodar desvios de frequência que existiam nos sistemas anteriores e a construção de equipamentos com sintonização rápida e precisa de qualquer canal em ondas curtas, *VHF* e *UHF*.

A integração em larga escala de circuitos rápidos, incorporou aos sintetizadores a possibilidade de serem programados para realizar um chaveamento de frequência em altas velocidades. Essa característica estendeu a aplicação de sintetizadores para vários circuitos e sistemas eletrônicos, com os quais, por exemplo, pode se obter a resposta em frequência de filtros na avaliação de sistemas de telemetria multicanal e para o uso em espalhamento de espectro.

Nos dias de hoje com a franca expansão dos sistemas micro controlados, citando a área de telecomunicações com a tecnologia *CDMA* (Code Division Multiple Access), notamos que o custo e velocidade de processamento de dados são determinantes para o sucesso de alguma determinada tecnologia ou para a implementação de um sistema [11].

No caso, por exemplo, dos dispositivos Bluetooth, os quais, para comunicação, utilizam uma frequência de rádio aberta, isto é, a faixa *ISM* (Industrial, Scientific, Medical), é necessário garantir que o sinal do dispositivo não sofra e não gere interferências.

O esquema de comunicação *FH-CDMA* (*Frequency Hopping – Code Division Multiple Access*), utilizado pelo Bluetooth, permite tal proteção, já que faz com que a frequência seja dividida em vários canais mudando de um canal para outro de maneira muito

rápida. Isso faz com que a largura de banda da frequência seja muito pequena, diminuindo sensivelmente as chances de uma interferência [20].

No contexto de elevada taxa de dados e comunicação em altas frequências, a Síntese Digital Direta oferece uma alternativa simples e eficiente, pronta para ser adicionada a uma arquitetura multi-função, embutida em um único chip notamos que o *DDS* é bastante eficaz.

2. Conceitos básicos sobre DDS

2.1. O que é DDS

A Síntese Digital Direta é uma técnica que utiliza um sinal de referência para definir intervalos de tempo, dentro dos quais são geradas amostras digitais de valores de uma senoide ou qualquer outra forma de onda. Cada amostra representa uma amplitude digital correspondente e estas por sua vez são convertidas por um conversor digital-analógico. Por último, um filtro linear elimina os resíduos da quantização, produzindo o sinal analógico de saída.

2.2. O sistema DDS

Ao contrário de grande parte da literatura, ao invés de começar com as formulações matemáticas, iniciaremos com a análise da Figura 1, a qual ilustra o sistema básico da Síntese Digital Direta:

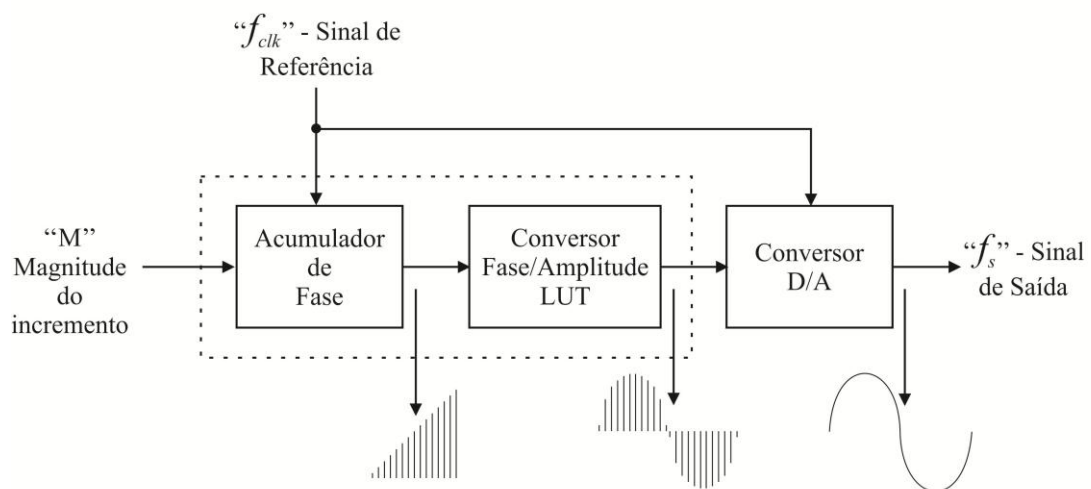


Figura 1: Esquema básico de um sistema de Síntese Digital Direta.

As entradas do sistema são:

- O Sinal de Referência, o qual definiremos por f_{clk} e é a nossa frequência de entrada do sistema, utilizada para sincronizar as diversas partes constituintes do sintetizador.
- A Magnitude do Incremento, a qual definiremos por M é um valor numérico que junto com a f_{clk} vai configurar o DDS para produzir as saídas desejadas.

Dentro dos três blocos básicos de implementação do método temos ainda:

- O Acumulador de Fase, o qual funciona como um contador, utilizando os dados fornecidos pela Magnitude do Incremento M e pelo Sinal de Referência f_{clk} para produzir amostras de fase, isto é, a cada ciclo de *clock* incrementa seu valor (de acordo com M) até seu valor máximo, retornando a zero para que o processo se repita.
- Conversor Fase/Amplitude, ou do inglês *Look-up-table LUT*, o qual em função de valores armazenados em sua memória (correspondentes a amostras de formas de ondas) converte o dado recebido do acumulador de fase para um valor correspondente de amplitude de cada amostra.
- Conversor digital-analógico (*DAC*), realiza a parte final do processo, que consiste, então, na conversão das amostras digitais em um sinal analógico contínuo no tempo.

Após a abordagem funcional e inicial do método vamos então retomar as etapas do processo introduzindo os conceitos matemáticos.

A senoide representada pela Figura 2 (a) pode ser expressa por $a(t) = \text{sen}(wt)$. É uma forma de onda difícil de ser gerada por circuitos analógicos. Representando a senoide pela projeção de um vetor girante Figura 2 (c), podemos usar como parâmetros a amplitude do vetor girante Figura 2 (b) e sua velocidade angular que é constante e expressa pela equação 1.

$$w = 2 \cdot \pi \cdot f_s$$

[1]

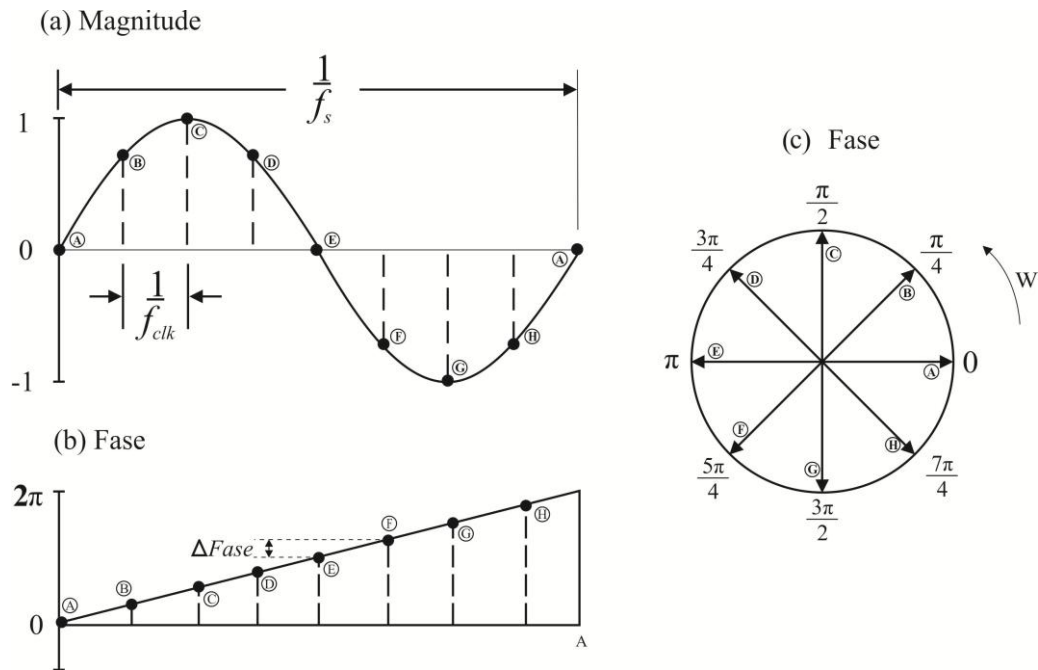


Figura 2: Magnitude da senoide e representações de fase

Sabendo que a informação de fase da senoide é linear, Figura 2 (b), e relacionada com a frequência de referência, a variação de fase $\Delta Fase$ por período pode ser determinada por:

$$\Delta Fase = wdt \quad [2]$$

Vamos, agora, apenas modificar a variável $\Delta Fase$ reescrevendo a equação 2 como:

$$M = wdt \quad [3]$$

A mudança de variável na equação 3, se justifica ao longo do trabalho pois no contexto do método essa variação de fase é diretamente relacionada a Magnitude do

Incremento, a qual, por esse motivo chamaremos de M (abordada em detalhes no capítulo 3.1). Resolvendo a equação 3 para w através da equação 1:

$$w = \frac{M}{dt} = 2 \cdot \pi \cdot f_s \quad [4]$$

Onde o valor mínimo de variação em relação ao tempo é:

$$\frac{1}{f_{clk}} = dt \quad [5]$$

Resolvendo a equação 4 para f_s e substituindo a variação da unidade de tempo pela frequência de referência f_{clk} na equação 5, especificamos a frequência do sinal de saída como:

$$f_s = \frac{M \cdot f_{clk}}{2 \cdot \pi} \quad [6]$$

Finalmente, para um acumulador de N bits, o sinal de saída terá a frequência especificada por:

$$f_s = \frac{M \cdot f_{clk}}{2^N} \text{ onde } 0 \leq M \leq 2^N \quad [7]$$

Um chip básico que implementa o *DDS* desenvolve a equação 7 usando os três blocos básicos (acumulador, *LUT* e conversor digital-analógico) descritos na Figura 1.

Pelo fato de que a forma de onda senoidal é um sinal determinístico, ou seja, pode ser descrito por uma função matemática, a síntese de frequência utilizada na estrutura é feita por meio de processamentos digitais e analógicos das amostras de uma onda senoidal.

Por esse motivo, podemos então simular a seqüência de passos descritos pela Figura 1 e que constituem o funcionamento do *DDS* com auxílio de uma ferramenta computacional.

A ferramenta computacional escolhida foi o *Matlab*® pois essa é muito popular e difundida na área de engenharia. Com ela é possível gerar as amostras contidas em um período completo de uma forma de onda senoidal com frequência, fase e amplitude determinadas, desde que o número e amostras calculadas seja finito. Essas amostras simuladas serão suficientes para o funcionamento da estrutura *DDS*, como será demonstrado.

A rotina implementada para o *Matlab*® (Apêndice – Código 1) permite que seja realizada a simulação do funcionamento básico do método. Definindo os parâmetros relacionados a configuração da quantidade de memória disponível na *LUT*, no acumulador de fase e a magnitude do incremento (explicitados na forma de comentário na rotina), podemos visualizar graficamente o comportamento das formas de onda nas etapas descritas pela Figura 1.

Os gráficos gerados abaixo (código no Apêndice – Código 1) são as respostas da rotina acima implementada, o primeiro, Figura 3, representa as amplitudes da senoide armazenada na *LUT*, cada endereço contém uma amplitude predefinida da onda senoidal, na configuração proposta pela simulação a senoide é armazenada em memória com 2^5 pontos.

O gráfico da Figura 4 nos mostra o acumulador de fase, o qual a cada ciclo de *clock* adiciona a magnitude do incremento (no caso da simulação $M = 4$) ao acumulador de fase.

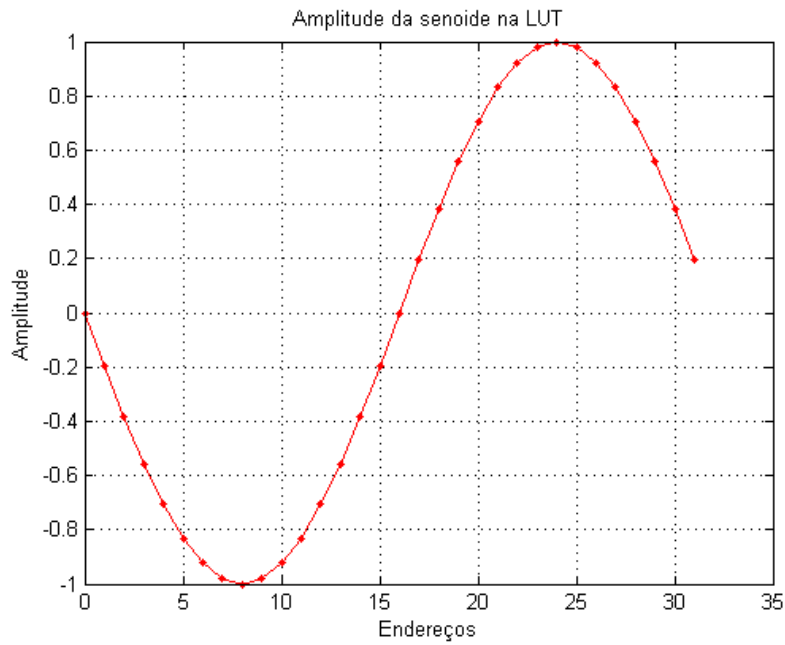


Figura 3: Amplitude da senoide armazenada na LUT.

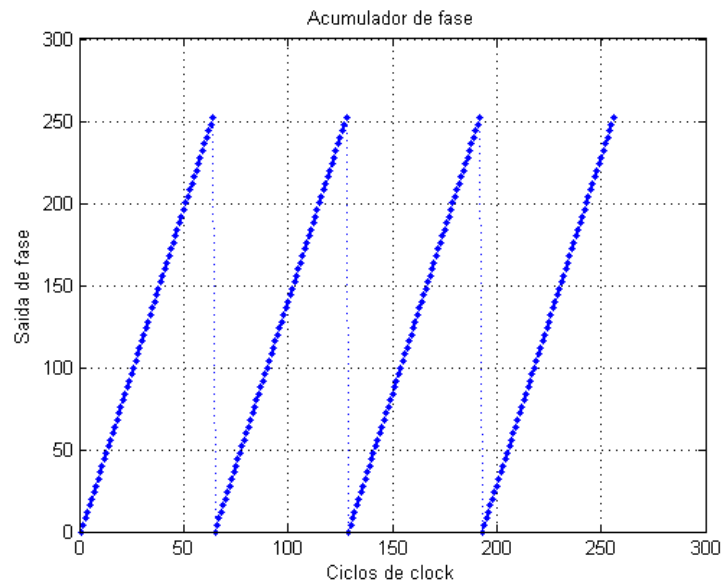


Figura 4: Saída digital de fase.

A Figura 5 ilustra a saída final do DDS. Cada ponto do gráfico representa a relação da saída digital de fase da Figura 4 com a correspondente amplitude da senoide armazenada na LUT.

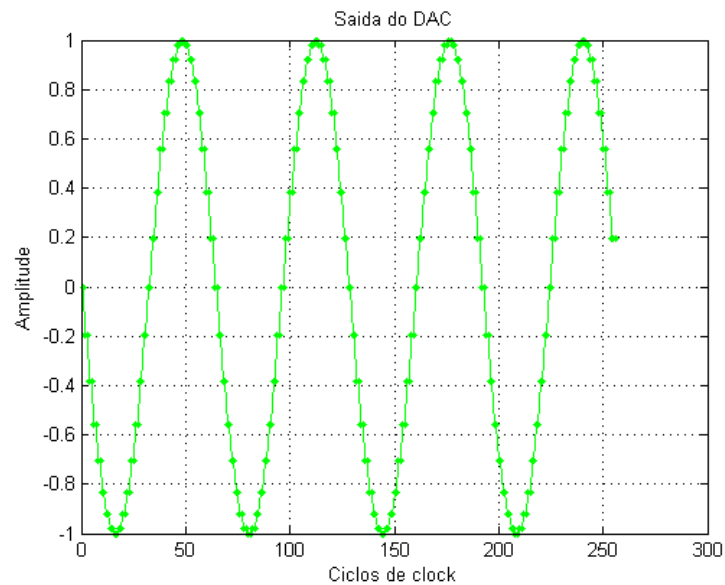


Figura 5: Saída do DAC.

3. Análise da arquitetura em blocos funcionais

A partir desse capítulo será abordada uma análise mais profunda e formal da arquitetura *DDS* aonde serão abordados também aspectos positivos e negativos.

3.1. Acumulador de Fase

Em primeira instância, como na Figura 6, podemos tratar o Acumulador de Fase como sendo um somador, ou seja, um módulo contador que incrementa o valor armazenado a cada pulso de *clock* recebido, pois, como visto no capítulo 2.2 o Acumulador de Fase utiliza o Sinal de Referência e a Magnitude do Incremento para atualizar o Registrador de Fase. Esse incremento, ou seqüência de valores, formam um laço cujos valores são incrementados até seu valor máximo retornando a zero para que o processo se repita novamente.

Devido a estas somas sucessivas teremos um sinal de varredura cuja magnitude do incremento será determinada pela palavra M . Fisicamente o acumulador de fase é constituído por um integrador digital, representado por um contador com incremento de fase variável.

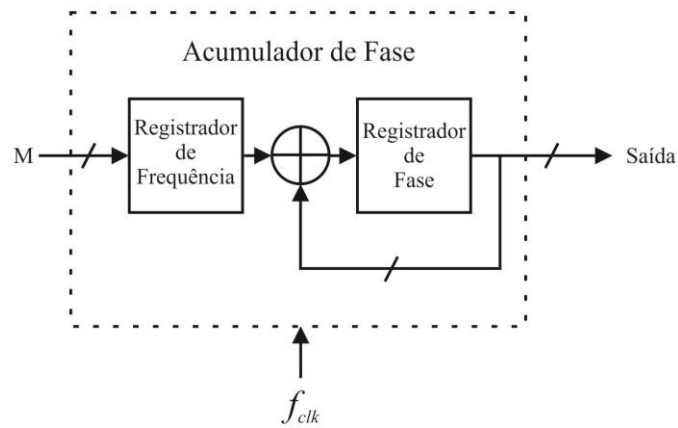


Figura 6: Célula mínima de um acumulador de fase.

A magnitude do incremento M guarda o tamanho do passo que será utilizado para a soma entre cada atualização do *clock*.

Uma abordagem que a *Analog Devices*® utiliza para explicar o funcionamento do acumulador de fases e a influência da palavra M no valor incrementado é utilizar o Círculo de Fases da Figura 7.

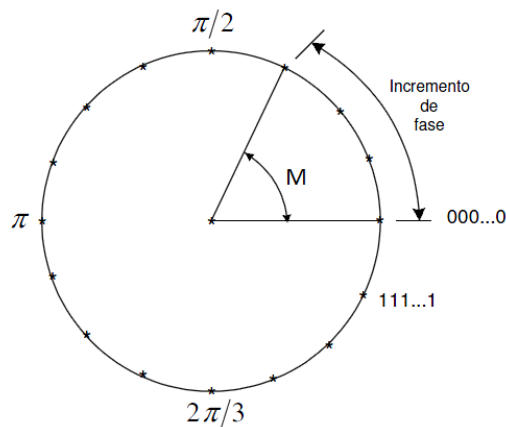


Figura 7: Círculo de fases.

Considerando a análise de um período de uma senoide com amplitude máxima unitária, o que fasorialmente nos leva a representação do círculo de raio unitário da Figura 7,

vamos interpretar as amostras de tempo como deslocamentos de fase, tal como visto no capítulo 2.2.

Então, a geração das amostras do sinal analógico corresponde ao processo de percorrer o círculo de raio unitário, no sentido anti-horário, com deslocamentos de fase fixos.

A cada ciclo de *clock* andamos uma fração do círculo (Figuras 2 e 7) e quanto maior este deslocamento, ou passo de fase, maior a frequência do sinal analógico produzido. Se um sinal senoidal contínuo no tempo tem ângulos de fase que se repetem na faixa entre 0 e $2\pi rad$, então na implementação digital podemos fazer analogia da oscilação da onda seno como sendo a extremidade do vetor com rotação ao longo do círculo, indicado na Figura 7, partindo do centro do círculo até a borda.

Fazendo algumas considerações sobre a palavra M , notamos que a partir do capítulo 2.2 até agora, ela foi citada basicamente como sendo a “magnitude do incremento”, entretanto após verificação do círculo de fases da Figura 7 vamos estender o conceito dessa variável para relacionar com outras nomenclaturas apresentadas na literatura. Outro nome freqüentemente utilizado é “incremento de fase” ou “variação de fase” quando interpretado como no capítulo 2.2. Quando se analisa a variável M do ponto de vista quantitativo, e em geral quando se está em fase de análise do sinal de saída, ela é comumente referenciada como “frequência da palavra de ajuste” ou simplesmente “palavra de ajuste”.

O número de pontos de fase discretos contido no círculo de fase é determinado pela resolução do acumulador de fase o qual é representado pela letra N , como o seu valor está contido na faixa entre 8 e 48, podemos ter ao longo do círculo de fase, como visto no capítulo 2.2, de 2^8 a 2^{48} pontos de fase discretos cuja representação se encontra na Tabela 1.

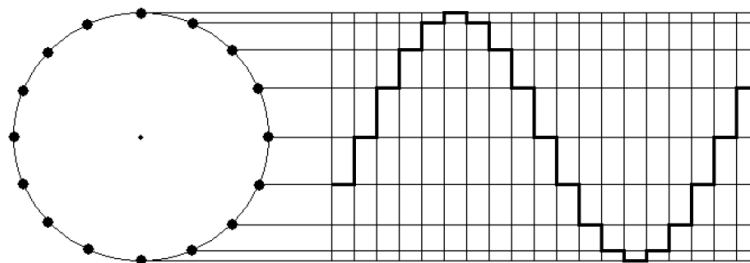
Tabela 1: Número de pontos do círculo de fase.

N	Número de pontos
8	256
12	4096
16	65535
20	1048576
24	16777216
28	268435456
32	4294967296
48	281474976710656

Quanto maior for o número de bits N do acumulador em mais partes o círculo de fase será dividido e maior será o número de informações do seno amostrado.

Para aumentar a faixa dinâmica de possibilidades de se excursionar o círculo em relação à amplitude da senoide amostrada, teremos que ter maior quantidade de bits para representar o número de pontos no círculo de fase N .

Sob outra perspectiva pode ser visualizado a correlação da roda de fases com a senoide na Figura 8. A quantidade de “degraus” na tabela é diretamente proporcional ao valor palavra de ajuste.

**Figura 8: Interpretação dos valores do círculo de fase.**

Como nos mostrou a equação 7, para cada ciclo completo da forma de onda gerada são necessários 2^N ciclos de máquina dividido pelo número da palavra de ajuste.

Como exemplo, para um acumulador de fase com 48 bits ($N=48$) usando sua precisão máxima, 34 bits são utilizados para armazenar a fase restante e 14 bits para escolher uma amostra da tabela de consulta. Com isso o acumulador de fase produz um endereço de 14 bits correspondente a fase exata do sinal. Para o exemplo em questão o endereço de 000...000 corresponde a 0° , por outro lado o endereço 111...111 corresponde a $359,978^\circ$. Assim o gerador de sinal pode utilizar esse endereço para controlar a fase do sinal a qualquer momento [7]. Isso é mostrado na Figura 9.

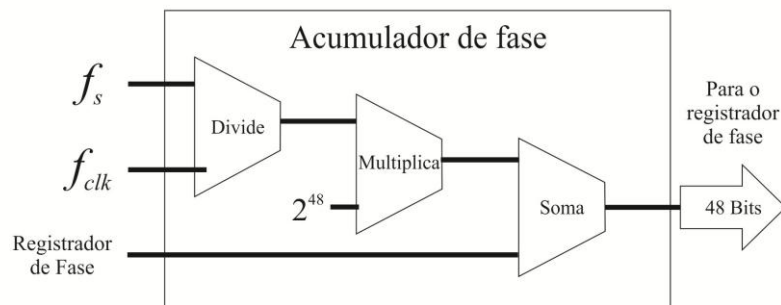


Figura 9: Computação do registrador de fase.

Como o diagrama da Figura 9 sugere, o acumulador de fase é capaz de representar a fase do sinal gerado com 2^{48} pontos de precisão. No entanto, pelo fato de que só temos 2^{14} pontos disponíveis na forma de onda, apenas os 14 bits mais significativos do registrador de fase serão utilizados na tabela de consulta. Os 34 bits restantes são utilizados para armazenar o restante do incremento de fase, este restante define a precisão do sistema de Síntese Digital Direta, assegurando que a tabela de consulta retorne a correta informação da fase após o registrador de fase ter “girado” [7].

Seja T o período do relógio principal (*clock*) e $2^N \cdot T$ o tempo necessário para se ler todos os endereços da tabela de consulta, então $2^N \cdot \frac{T}{M}$ é o tempo necessário para se ler algum

endereço específico, determinados pelo incremento de endereço M . Podemos interpretar esse tempo como o período do sinal sintetizado de saída dado pela equação 8 [2].

$$f_s = \frac{M}{T \cdot 2^N} = \frac{M \cdot f_{clk}}{2^N} \quad [8]$$

Através da Figura 7 e da equação 8 pode-se perceber a relação entre o número de bits do acumulador de fase, a precisão e a resolução do sinal sintetizado.

Ainda em consideração a equação 8 e ao considerar o menor valor para a palavra M , ou seja, $M = 1$ pois esse deve ser um valor inteiro e maior que zero, podemos considerar o número de bits N do acumulador de fase como o responsável por definir a resolução do sinal sintetizado de saída, ou seja, a variação de $\frac{f_{clk}}{2^N}$ que provoca uma variação perceptível no sinal de saída. Chegando então na equação 9:

$$f_s = \frac{f_{clk}}{2^N} \quad [9]$$

Podemos ainda reescrever a equação 8 em termos da frequência f_s , como função de M [5]:

$$f_s = f_s(M) = \frac{f_{clk}}{2^N} \cdot M \quad [10]$$

O incremento de fase M , arredondado para o número inteiro mais próximo, é dado pela equação 11:

$$M = \left[f_s \cdot \frac{2^N}{f_{clk}} + 0,5 \right] \quad [11]$$

A frequência mínima de saída $f_{s_{min}}$ que o *DDS* pode gerar, ou em outras palavras, o menor valor da palavra de ajuste que ainda consegue incrementar a fase (pois $M = 0$ não incrementa a fase), como foi visto na equação 10 com $M = 1$ é:

$$\begin{aligned}\Delta f_{s_{min}} &= f_s \cdot (M + 1) - f_s \cdot (M) \\ \Delta f_{s_{min}} &= \frac{f_{clk}}{2^N} \cdot (M + 1 - M) \\ \Delta f_{s_{min}} &= \frac{f_{clk}}{2^N}\end{aligned}\quad [12]$$

A forma da equação 12 permite que o projetista escolha, a partir do número de bits N e do acumulador M a variação mínima na frequência de saída. O tamanho da palavra de ajuste de frequência N ainda pode ser reescrita como [5]:

$$N = \left\lceil \log_2 \cdot \left(\frac{f_{clk}}{\Delta f_{s_{min}}} \right) + 0,5 \right\rceil \quad [13]$$

A frequência máxima $f_{s_{max}}$ que o *DDS* pode gerar é dada pela amostragem segundo o teorema de amostragem de Nyquist equação 14, que será abordada com mais detalhes no capítulo 5.3 [5].

$$f_{s_{max}} = \frac{f_{clk}}{2} \quad [14]$$

Para um ponto de vista prático a frequência $f_{s_{max}}$ pode ser comumente calculada como sendo $f_{s_{max}} = \frac{f_{clk}}{4}$ pois frequências de saída mais baixas do que essa são facilmente reconstruídas utilizando um filtro passa baixas.

3.2. Tabela de conversão *LUT* / Conversor fase para amplitude

De acordo com descrições anteriores, o acumulador de fase será responsável por endereçar a leitura da tabela de conversão. Dentro dela estarão armazenados todos os dados referentes a um período da onda senoidal gerada, como foi visto no capítulo 2.2 na simulação com o *Matlab*®.

Quanto maior for o número de bits do acumulador de fase, maior será a quantidade de dados armazenados na *LUT*. Observando a Figura 7 e entendendo que na *LUT* estarão armazenadas somente as amostras referentes ao conteúdo de um ciclo de fase completo 2π , fica evidente que dependendo do incremento de fase escolhido serão lidos mais ou menos endereços da *LUT*, com um tempo maior ou menor. Este tempo é proporcional ao incremento de fase M , visto na equação 9 [2].

Esses valores buscados na tabela *LUT*, serão por sua vez os valores de amostras das formas de onda (triangular, senoidal, quadrada, ou até mesmo algum tipo de onda pré determinado pelo usuário) que serão então enviadas ao conversor digital-analógico que fará a última conversão antes de fornecer o sinal para o filtro passa-baixas na saída.

Na Figura 10 se ilustra dois sinais sintetizados com tamanhos de palavra diferentes porém com mesmo sinal f_{clk} , $M = 1$ e $M = 2$ [2].

Para o momento em que $M = 1$, como o incremento é unitário, o acumulador de fase varrerá todos os possíveis endereços da *LUT*, gerando a menor frequência sintetizada possível. Tem-se o maior tempo de leitura da *LUT* e, conseqüentemente, o maior período possível para uma senoide sintetizada pela estrutura da *DDS*. Para o caso de $M = 2$, temos a metade do período da senoide do exemplo anterior, isto acontece porque o acumulador de fase do segundo exemplo passará pela metade dos endereços para varrer o mesmo ciclo de fase completo da senoide amostrada [2], ou seja, só varrerá os endereços pares da *LUT*. Assim fica

claro o controle da frequência do sinal sintetizado através do controle do incremento de fase do acumulador [3].

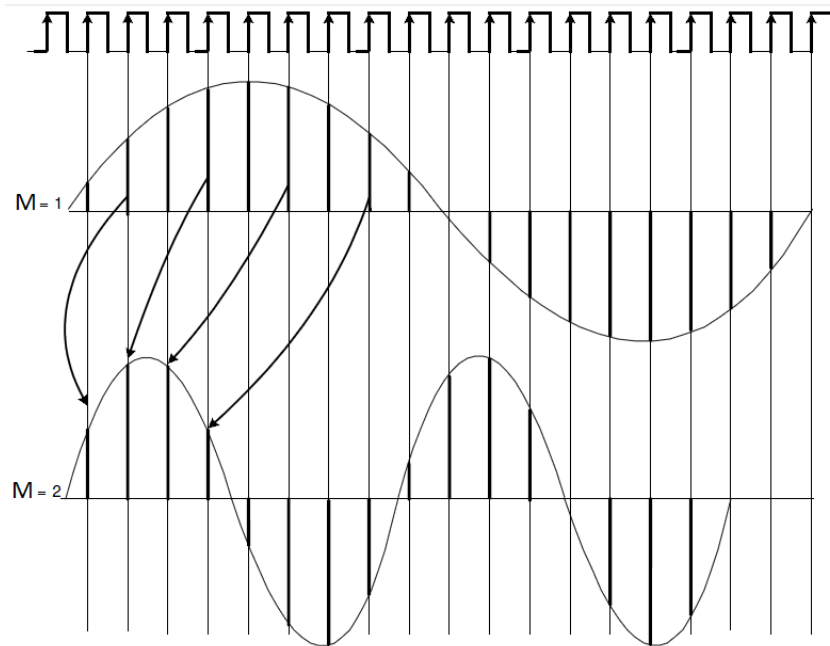


Figura 10: Configuração para M=1 e M=2.

Na Figura 10, a primeira amostra do sinal de $M = 2$ é equivalente a segunda amostra de $M = 1$, a segunda amostra de $M = 2$ é equivalente a quarta amostra de $M = 1$ e assim sucessivamente.

Uma fórmula que também pode ajudar no entendimento do conceito da variação da frequência do sinal sintetizado de saída, mudando-se apenas o incremento de fase M é [2]:

$$m = \frac{d\phi}{dt} \quad [15]$$

Sendo m a frequência angular e ϕ a fase do sinal. A frequência do sinal sintetizado está relacionada com a taxa de variação da sua fase no tempo. Na estrutura *DDS*, nota-se que pela variação do incremento de fase M é possível modificar a frequência do sinal sintetizado.

Um ponto interessante de ressaltar nesse momento é que na arquitetura *DDS* ao contrário de outras que utilizam osciladores analógicos essa mudança de frequência do sinal sintetizado pode ser adquirida instantaneamente (1 ciclo de f_{clk}) sem necessidade de um tempo maior de acomodação.

Quanto maior o valor de M maior será a inclinação da variação da fase, maior será o resultado da derivada da fase em relação ao tempo e maior o valor da frequência sintetizada, confirmando a relação de compromisso da saída de fase Figura 4 com um valor maior de M .

Um fator que deve ser considerado é a geração da modulação parasita de fase (*Jitter*), correspondente à palavra de controle M escolhida. Este fato acontece devido ao comportamento do acumulador de fase.

Em uma linguagem informal, o jitter é um tipo de oscilação no instante de tempo em que é esperada a ocorrência de um evento qualquer.

Em um sistema digital de transmissão ideal a duração dos pulsos de um sinal é sempre a mesma ao longo do tempo. Porém, no caso real, a duração desses pulsos variam. Essas variações são modulações indesejáveis definidas pelo jitter. Assim, o jitter é uma medida do quanto o sinal oscilou em relação à sua posição ideal.

A preocupação com o jitter de fase justifica-se pelo impacto causado no desempenho de sistemas de comunicação. Quando o jitter é excessivo, dados podem ser perdidos durante o processo de transmissão ou interpretados de maneira errada, principalmente quando codificados por algum método baseado em variação de frequência ou de fase. Os primeiros sistemas de comunicação eram analógicos, transmitindo dados a taxas relativamente baixas. A influência do jitter era desprezível. A demanda por serviços de alta velocidade levaram ao desenvolvimento tecnológico desses sistemas, que passaram a transmitir dados em sinais digitais a velocidades cada vez maiores. Neste cenário a degradação provocada pelo jitter é significativa. Por este motivo é importante entender o fenômeno do jitter e identificar as duas

diversas causas. Modelar o jitter matematicamente permite que seu efeito no sistema seja atenuado através da identificação e do ajuste dos parâmetros envolvidos.

A terminologia envolvendo o jitter não é padronizada, ou seja, é comum encontrar na literatura termos como jitter, jitter de fase, timing jitter, jitter periódico, entre outros. Por vezes, tipos diferentes de jitter são descritos pelo mesmo termo, ou então, um certo jitter admite várias denominações [25].

Realizando alguns testes é possível notar que existem alguns valores para a palavra de controle M que são divisores exatos de 2^N . Para estes, o sinal de saída do acumulador de fase apresenta um comportamento perfeitamente periódico. Para outros valores que a palavra M assume, não divisores exatos de 2^N , nota-se uma quebra na periodicidade do sinal de sua saída e o aparecimento de um *Jitter* indesejado. A Figura 11 considerando uma memória com $N = 4$ bits de endereçamento, onde $2^N = 16$.

Pelo fato de que funções como “ $\sin x$ ” não são lineares a implementação com componentes discretos é praticamente inexistente, quase na totalidade das vezes são utilizados dispositivos de armazenamento, esses dispositivos, como já citados, armazenam os valores de amostras da forma de onda que será entregue ao conversor digital-analógico [2].

A Figura 12 exemplifica uma memória *ROM* ideal, notamos que para representar idealmente uma onda senoidal precisaríamos de uma resolução infinita [2].

M = 2 → 0 2 4 6 8 A C E 0 2 4 6 8 A C E 0 ...
 M = 6 → 0 6 C 2 8 E 4 A 0 6 C 2 8 E 4 A 0 ...

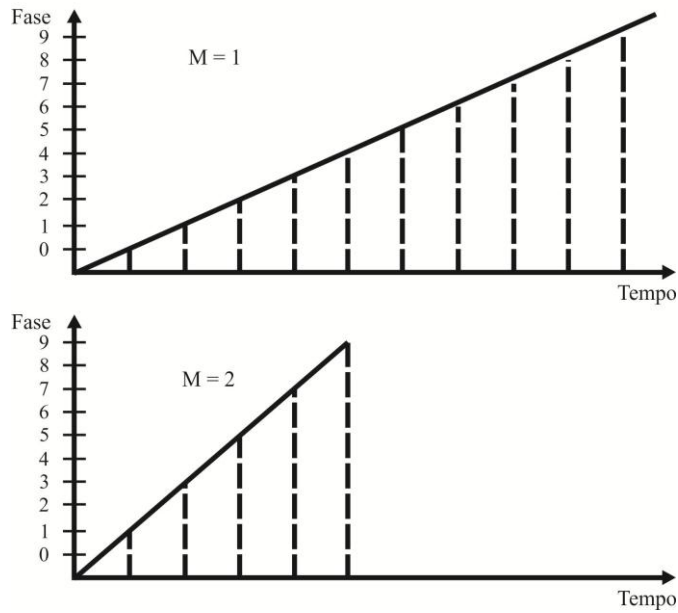


Figura 11: Relação entre o incremento de fase e a taxa de variação da fase.

O sinal de saída poderia ter resolução e precisão máximas na frequência sintetizada e menor geração de espúrios devido ao ruído de quantização ser nulo para este bloco. Como este caso não é possível e nem viável, é necessário conviver e projetar uma redução no



Figura 12: Memória ideal (infinita).

número de bits da memória, suficiente para que o desempenho da estrutura *DDS* não seja comprometida.

É interessante lembrar que o tipo da memória está diretamente ligado às suas características de velocidade, consumo, tamanho e de preço, fatores importantes em um projeto. Visando otimizar o tamanho da memória *ROM* (memória da *LUT*) e minimizar os efeitos do truncamento do número de bits que repercute em um “ruído de quantização”, foram implementados vários métodos de compressão de memória (abordado no capítulo 4.1). Estes são capazes de diminuir o conteúdo da memória *ROM* sem que haja perdas na qualidade do sinal sintetizado [2].

3.3. Conversão digital-analógico

O conversor digital-analógico realiza a conexão entre o processamento digital e o processamento analógico do sinal sintetizado. Ele relaciona cada palavra digital da sua entrada a um nível de tensão que varia entre os limites de sua tensão de alimentação V_{CC} , ou seja, a resolução do *DAC* é especificada pela quantidade de bits de entrada e a máxima amplitude do sinal pela tensão de alimentação. Por exemplo, a resolução de um *DAC* com 10 bits de entrada é referenciada como “resolução de 10 bits”.

O impacto da resolução do *DAC* é mais facilmente entendido observando a Figura 13 na qual um conversor de quatro bits (traço quantizado preto) é utilizado para reconstruir uma onda senoidal perfeita (traço vermelho). As linhas verticais são marcas de tempo e identificam os instantes de tempo nos quais a saída do conversor é atualizada para um novo valor. As distâncias horizontais entre as linhas verticais representam o período da amostra [3].

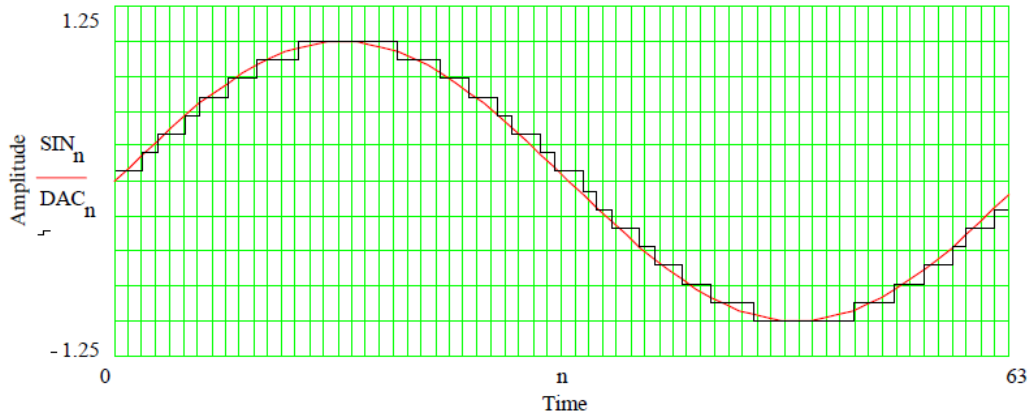


Figura 13: Resolução do DAC.

Tomando outro exemplo, se um conversor for alimentado com 2V de alimentação simétrica, ou seja, de -1V a 1V e trabalhar com 10 bits, o sinal em sua saída pode variar de -1 a 1 volt com um passo de $\frac{2}{2^{10}}$ volts, como observado na Figura 13.

Podemos observar com mais clareza todas as etapas do processo descritas na Figura 14.

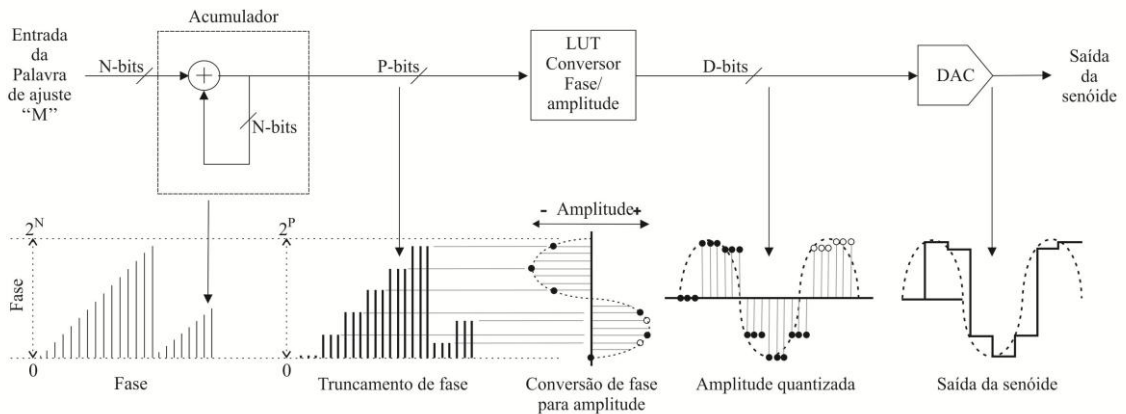


Figura 14: Esquema geral resumido da síntese

3.4.Considerações técnicas adicionais

Quando nos referimos à tecnologia e velocidade de processamento o objetivo maior sempre é a otimização, ou seja, devemos conseguir otimizar o tamanho de memória, consumo de energia e velocidade de dados a serem processados.

Entretanto, para fazermos tais análises devemos em primeiro lugar definir os tamanhos máximo e mínimo de cada palavra que irá configurar e ser responsável pelo funcionamento dos blocos básicos que implementam o *DDS*. Examinando a literatura a quantidade de bits disponíveis para cada tamanho de palavra não é fixa e varia conforme o fabricante e proposta de funcionalidade de cada implementação.

Essa variação em relação a alocação dos bits se deve ao fato de que os fabricantes pressupõe algumas aplicações típicas para a utilização do *DDS*.

Como, de um modo geral, para aplicações industriais apenas os módulos básicos não são suficientes para fornecer a qualidade e quantidade de dados que são requeridos, os módulos de implementação podem ainda integrar outras funções como filtro passa baixas, métodos de compressão de memória e até módulos de gerações de formas de onda pseudo-aleatórias.

Para conseguirmos efetivamente extrair o máximo da arquitetura, dentro dessa implementação básica a qual disponibiliza poucos recursos precisamos explorar alguns pontos limites de operação.

Tais pontos limites ao serem explorados acabam gerando efeitos específicos que se apresentam de formas diferentes a cada etapa do processo.

Muitos desses pontos fogem ao escopo deste trabalho, pois o propósito geral não é explorar apenas uma arquitetura e sim o funcionamento geral do método.

A fundamentação teórica aqui oferecida fornec os subsídios para escolher dentro das arquiteturas disponíveis no mercado, uma que seja pelo menos suficiente para agregarmos a nossa aplicação de interesse. Assim, será abordado de forma superficial apenas qualitativa, os fenômenos envolvidos e a disponibilidade mais comumente oferecida pelo mercado em relação ao tamanho de palavras e suas implicações.

Com essas considerações caso seja de interesse, o leitor tem o embasamento e a motivação para aprofundar a análise de algum ponto específico.

Primeiramente vamos então definir os dois pontos básicos que serão utilizados para nossa escolha, são eles:

- Faixa dinâmica: É a faixa de amplitude de operação do sinal dentro de uma região de trabalho, ou em outras palavras, a relação entre o valor máximo e mínimo do sinal fornecido. Os níveis são medidos em decibel [dB] .
- Resolução de frequência: É a menor variação de frequência que pode ser diferenciada dentro da faixa dinâmica do sinal.

Com o intuito de escolhermos a quantidade de bits necessários para a *LUT* devemos estipular, primeiramente, qual a faixa dinâmica necessária no nosso projeto. Para o exemplo de uma tabela de consulta com 22 bits, o sinal de saída (com amplitude de pico a pico = 2V) será diferenciado com amplitudes equivalentes a $\frac{2}{2^{22}}$ do valor total da faixa dinâmica.

Para definirmos qual frequência máxima e mínima de saída, devemos analisar o comportamento da equação 7 em relação a variação da palavra de ajuste *M*.

Para determinar a frequência mínima devemos analisar a variação da equação 12 em relação a variação de bits *N* do acumulador de fase, contudo para análise do tamanho de *N* porém com base apenas nos comportamentos de *clocks* podemos trabalhar com a equação 13.

Para a frequência máxima sem considerar o critério de Nyquist podemos levar em consideração a equação 14.

As palavras abaixo discriminadas foram levantadas levando em consideração a disponibilidade comercial dos CI's de DDS. Considerando desde protótipos construídos em larga escala pela China com o propósito de serem simples e baratos até disposições muito mais robustas como placa de testes da Analog Devices por exemplo.

Tamanho das palavras:

- **Resolução de fase do acumulador “*N*”:**

Disponibilizado pelos fabricantes com valores entre 8 e 48 bits.

- **Tamanho da palavra de ajuste “*M*”:**

Disponibilizado pelos fabricantes com valores entre 24 e 48 bits.

O tamanho da palavra escolhida pode repercutir na modulação parasita de fase *Jitter* que é o deslocamento dinâmico das arestas de sinais digitais em relação as suas posições médias, medida em graus *RMS*.

- **Resolução de amplitude da *LUT* “*L*”: bits**

Disponibilizado pelos fabricantes geralmente com 22 bits.

Valores de resolução de amplitude muito elevados em relação a resolução do conversor digital-analógico podem repercutir em níveis de ruído elevados.

- **Resolução de fase da *LUT* “*P*”:**

Disponibilizado pelos fabricantes com valores entre 14 e 16 bits.

Contudo ainda são executados métodos de compressão de memória como por exemplo a “simetria do quarto de onda” e o truncamento de fase. O truncamento de fase quando repercute em erro, pode contribuir para aparecimento de ruídos espúrios “ruído de quantização”.

- **Resolução do *DAC* “*D*”:**

O valor da quantidade de bits disponíveis para o conversor digital-analógico é um dos pontos que mais variam em relação a disponibilidade dos fabricantes. É encontrado na faixa de valores entre 8 e 32 bits.

4. Implementação prática - Simulações

A implementação das rotinas em *Matlab*® e respectivas saídas gráficas, adaptadas pelo autor, descritas a seguir, permitem que após todo o embasamento teórico seja possível explorar de um modo rápido e fácil, o impacto de cada parâmetro e se consiga alcançar a melhor decisão para escolher entre as arquiteturas e ainda comparar os resultados práticos com teóricos.

Os parâmetros que são explicitados nas descrições de cada gráfico podem ser alterados facilmente através de algumas pequenas mudanças nas rotinas que se encontram comentadas no Apêndice.

Além dos itens que são especificados na descrição dos gráficos ainda podemos ajustar o período de interesse e amplitude de visualização de cada sinal. Esses ajustem também se encontram comentados no código.

Algumas das simulações contemplam assuntos que ainda não foram abordados até o presente momento, contudo se referem ao próximo capítulo (capítulo 5) o qual explora algumas características bem específicas. Estes códigos, assim como os demais, se encontram comentados no Apêndice.

4.1.Saída digital de fase

O gráfico da Figura 15 simula o comportamento da saída digital de fase, é o comportamento da primeira forma de onda da Figura 14. Os parâmetros utilizados para configuração foram, $f_{clk} = 20$ MHz, $N = 22$ bits, $M = 2^{18}$ bits e $T = 0,8 \mu s$. Esses parâmetros descritos podem ser facilmente ajustados na simulação. O código comentado para esse objetivo se encontra no Apêndice – Código 2 [21].

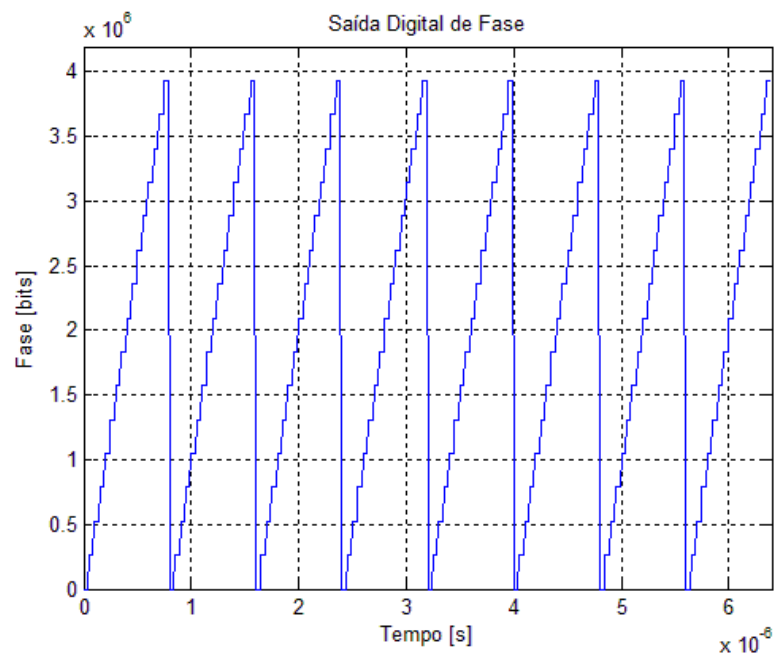


Figura 15: Saída digital de fase caso 1

O gráfico da Figura 16 também simula o comportamento da saída digital de fase. Também é baseado no Código 2 porém teve as variáveis ajustadas para simulação com, $f_{clk} = 20$ MHz, $N = 22$ bits, $M = 2^{16}$ bits e $T = 0,32 \mu\text{s}$.

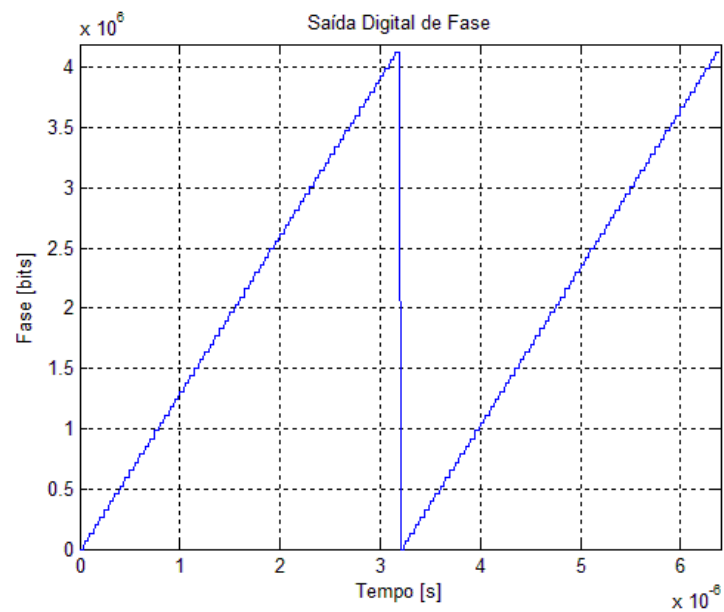


Figura 16: Saída digital de fase caso 2

4.2.Saída do conversor digital-analógico

O gráfico da Figura 17 simula o comportamento da saída do conversor digital-analógico. É o comportamento da última forma de onda da Figura 14. Os parâmetros utilizados para configuração foram, $f_{clk} = 20$ MHz, $N = 22$ bits, $M = 2^{18}$ bits e $T = 0,8 \mu s$. Da mesma forma que os gráficos anteriores podemos ajustar os parâmetros descritos. O código comentado para esse objetivo se encontra no Apêndice – Código 3 [21].

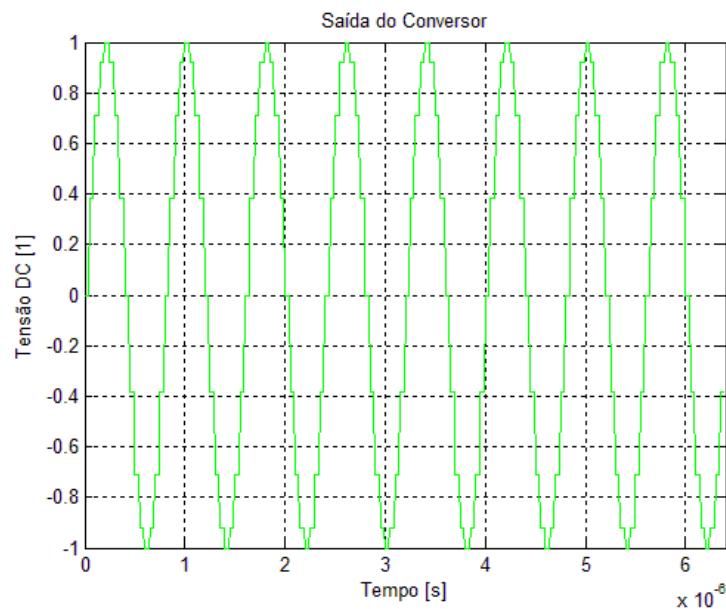


Figura 17: Saída do conversor digital-analógico caso 1

O gráfico da Figura 18 também simula o comportamento do conversor digital-analógico, é baseado no Código 3 porém teve as variáveis ajustadas para simulação com, $f_{clk} = 20$ MHz, $N = 22$ bits, $M = 2^{16}$ bits e $T = 0,32 \mu s$.

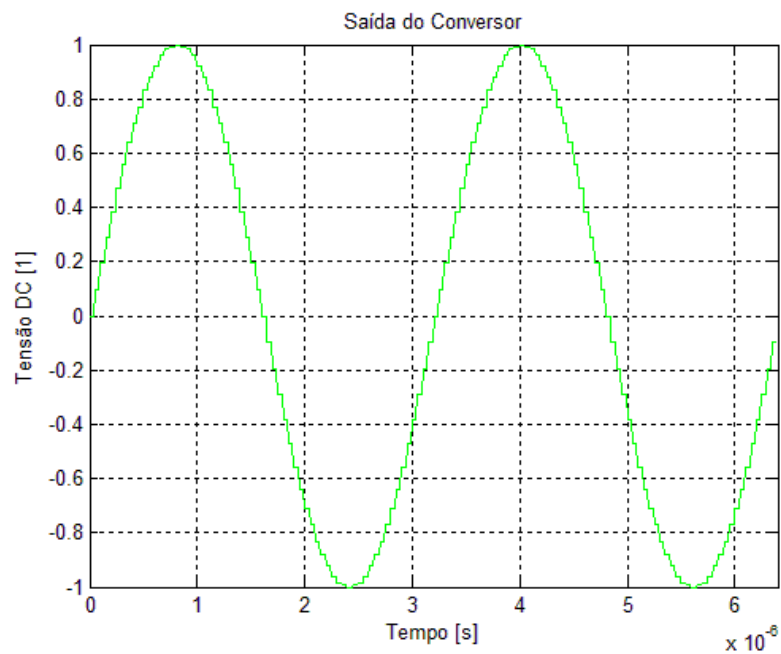


Figura 18: Saída do conversor digital-analógico caso 2

4.3. Saída do conversor digital-analógico com filtro passa-baixa

O gráfico da Figura 19 simula o comportamento da saída do conversor digital-analógico porém com a inclusão de um filtro Butterworth de ordem 2 no final. É o comportamento da última forma de onda da Figura 14 porém suavizado pelo efeito do filtro.

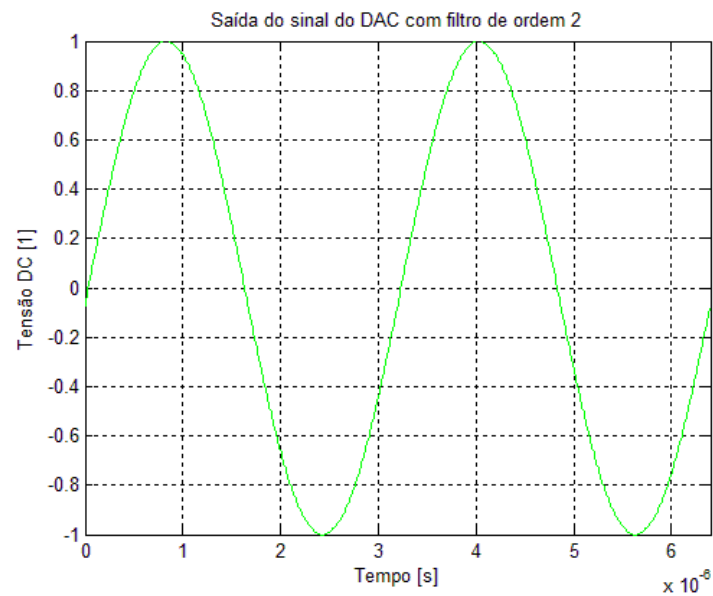


Figura 19: Saída do conversor digital-analógico com filtro de ordem 2

Os parâmetros utilizados para configuração e que podem ser ajustados são: $f_{clk} = 20$ MHz, $N = 22$ bits, $M = 2^{16}$ bits e $T = 0,32 \mu s$. O código se encontra no Apêndice – Código 4 [21].

4.4.Efeito do ruído de quantização do sinal do conversor digital-analógico

O gráfico da Figura 20 simula o comportamento do ruído de quantização de saída do conversor digital-analógico, será abordado em sugestão para trabalhos futuros . Os parâmetros utilizados para configuração e que podem ser alterados são: $f_{clk} = 20$ MHz, $N = 22$ bits, $M = 2^{16}$ bits e $T = 0,32 \mu s$. O código se encontra no Apêndice – Código 5 [21].

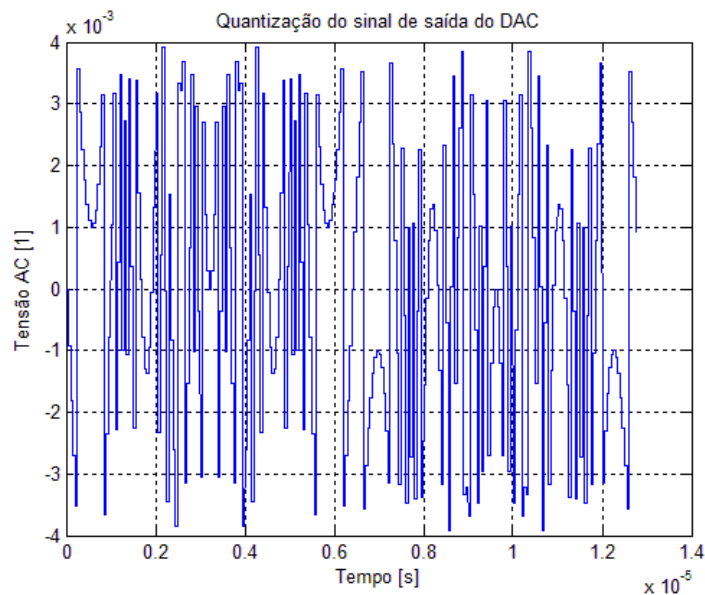


Figura 20: Efeito do ruído de quantização do sinal de saída do conversor digital-analógico

4.5.Espectro do sinal de saída do conversor digital-analógico

O gráfico da Figura 21 simula o comportamento do espectro de saída do conversor digital-analógico. Os parâmetros utilizados para configuração e que podem ser alterados são: $f_{clk} = 20$ MHz, $N = 22$ bits, $M = 2^{16}$ bits e $T = 0,32 \mu s$. O código se encontra no Apêndice – Código 6 [21].

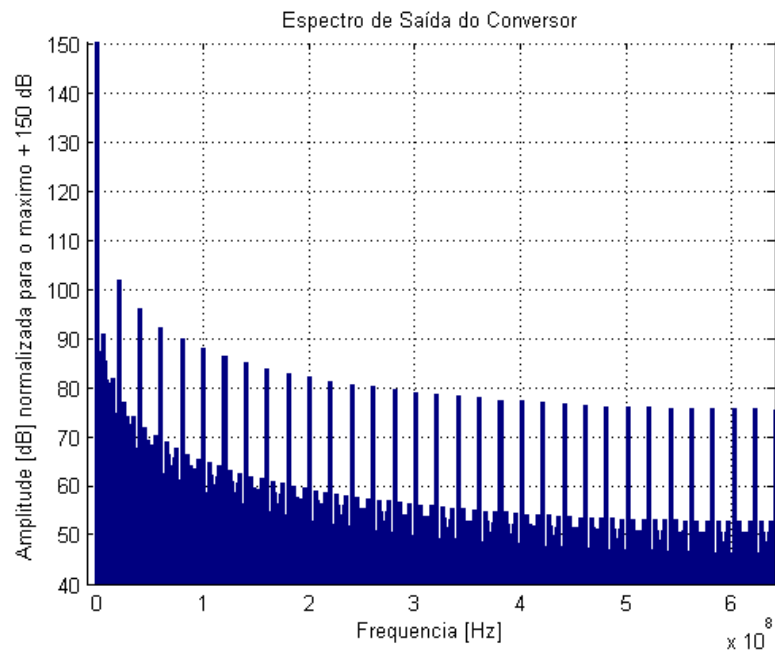


Figura 21: Espectro do sinal de saída do conversor digital-analógico

5. Trabalhos futuros

A seguir, algumas características da arquitetura *DDS* mencionadas nos capítulos anteriores serão examinadas mais profundamente. Esse capítulo tem como público alvo estudantes da área de engenharia. Como muitos dos assuntos aqui tratados são de certa forma relativamente complexos, um estudo aprofundado de cada ítem demandaria pelo menos um capítulo de análises. Por tal motivo, e como considero de grande importância para o público a qual destino este capítulo, insiro algumas informações para direcionar trabalhos mais profundos.

5.1. Truncamento de fase / bits

Considere um sistema *DDS* com um acumulador de fase 32 bits e um conversor digital-analógico de 8 bits Figura 22 [3]. Se o design implementado mantiver todos os bits de resolução então o núcleo do *DDS* ocupará uma área extremamente grande de memória e dissipará uma potência relativamente alta. Para convertermos diretamente os 32 bits de fase para uma amplitude correspondente, necessitaríamos de 2^{32} entradas na *LUT*, ou seja, 4.294.967.296 entradas. Se cada entrada é armazenada com 8 bits de precisão, então seria necessário 4 Gigabytes de memória na tabela de consulta para atender a demanda de resolução.

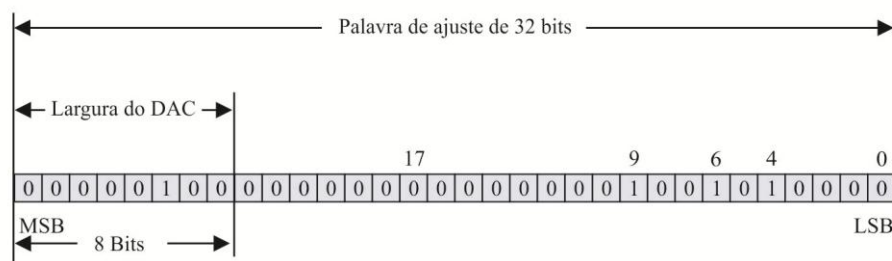


Figura 22: Palavra de ajuste de 32 bits.

No entanto, considerando o custo, tamanho, consumo de energia e velocidade de operação seria inviável implementar tal projeto. A arquitetura *DDS* é forçada a reduzir o tamanho da *LUT*.

Truncando os valores do acumulador de fase, ou seja, considerando apenas os bits mais significativos armazenados no acumulador para mapearem a conversão de fase para amplitude, conseguimos reduzir a dissipação de potência e conseqüentemente reduziremos o mapeamento de fase para amplitude.

Como resultado, alguns bits, os menos significativos (*LSB*) da saída do acumulador de fase são descartado ou truncados, assim, apenas os bits mais significativos da saída do acumulador fornecem as informações de fase, Figura 23.

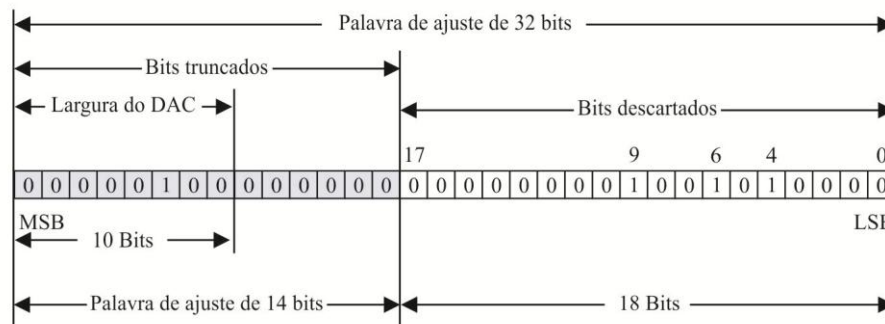


Figura 23: Palavra de ajuste de 32 bits com truncamento de fase.

Essa redução do tamanho da *LUT* leva a um erro de truncamento de fase o qual contribui para o aparecimento de ruído espúrio. Vamos analisar como esse ruído aparece através de um exemplo simples (poucos bits).

Considere uma arquitetura *DDS* simples que usa um acumulador de 8 bits junto a uma *LUT* de 5 bits (apenas os 5 bits mais significativos são utilizados para armazenar a informação de fase). O círculo de fase desse exemplo está ilustrado na Figura 24.

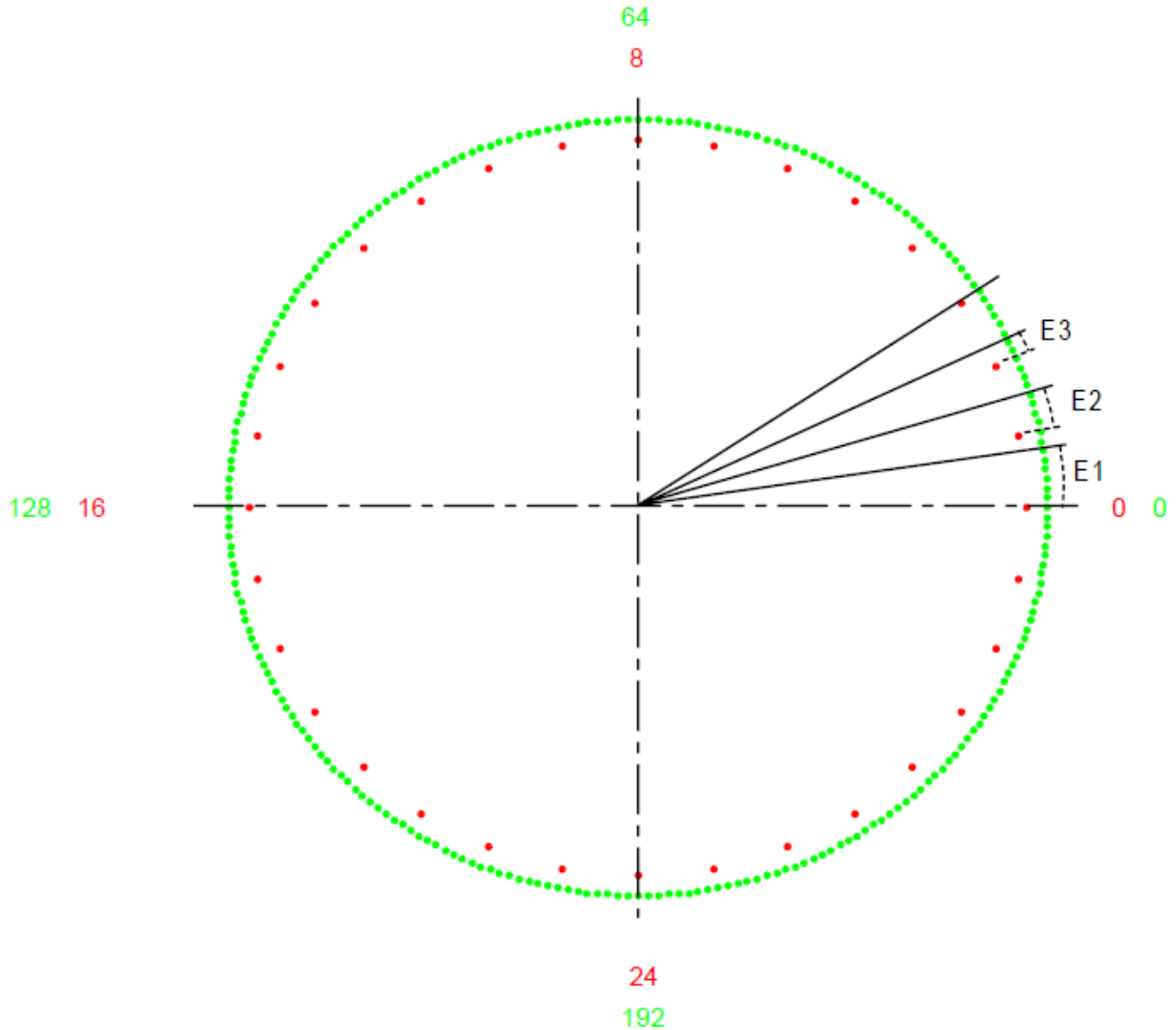


Figura 24: Erro de truncamento de fase e roda de fase

Com um acumulador de 8 bits ($N = 8$), a resolução de fase associada ao acumulador é $1/256$ de um círculo completo, ou $1,41^\circ$ ($360^\circ/256$). Na Figura 24 [3], a resolução do acumulador de fase é identificado pelo círculo externo marcado com bolinhas verdes. Se apenas os 5 bits mais significativas do acumulador são utilizados para transmitir a informação de fase ($P = 5$), então a resolução torna-se $1/32$ de um círculo completo, ou $11,25^\circ$ ($360^\circ/32$). Estes são identificados pelo círculo interno com bolinhas vermelhas.

A redução de memória passou de 256 (2^N) para 32 (2^P) endereços. Quando programada uma frequência com palavra de ajuste com valor $M=4$, a amplitude de leitura do

DAC pula três endereços da *LUT* a cada ciclo de *clock*. Por conseqüência, o *DDS* usa apenas 8 das 32 amplitudes de saída para construir a senoide.

$$f_o = \frac{4 \cdot f_c}{2^8} = \frac{f_c}{64} ; \text{ com } M = 4 \quad [16]$$

Agora vamos supor que uma palavra de ajuste de tamanho $M = 6$ seja utilizada. Então o acumulador contará através de incrementos de 6 unidades. Os primeiros quatro ângulos de fase correspondendo a 6 unidades de “passo” do acumulador estão representados na Figura 17. Note-se que o primeiro passo de fase (6 contagens no círculo externo) se localiza “dentro” da primeira marca interna. Assim, surge uma discrepância entre a fase do acumulador (o círculo externo) e a fase determinada pelos 5 bits de resolução (o círculo interno). Esta discrepância resulta em um erro de fase de $8,46^\circ$ ($6 \times 1,41^\circ$), como descrito pelo arco E1 na Figura 24 [3].

No segundo passo de fase do acumulador (próximas 6 contagens no círculo externo) a fase do acumulador reside entre a 1ª e 2ª marcas no círculo interno. Novamente, existe uma discrepância entre a fase do acumulador e a fase determinada pelos 5 bits de resolução. O resultado é um erro de $5,64^\circ$ ($4 \times 1,41^\circ$) como descrito pelo arco E2 na Figura 17.

Analogamente, no 3 passo de fase do acumulador um erro de $2,82^\circ$ ($2 \times 1,41^\circ$) resulta. No 4 passo de fase, no entanto, o acumulador de fase e fase dos 5 bits de resolução, coincidem sem nenhum erro de fase. Este padrão continua com incrementos de 6 unidades no acumulador contados pelo círculo externo.

Podemos visualizar o comportamento do truncamento de fase nos exemplos das Figuras 25 e 26 [17]:

Nas Figuras 25 e 26 a *LUT* com utilização total é utilizada como referência. A saída do acumulador de fase não é truncada e a saída do conversor digital-analógico é mostrada como

sendo uma senoide “suave”. As configurações dessas primeiras formas de onda são ($N=8$; $L=8$; $M=4$).

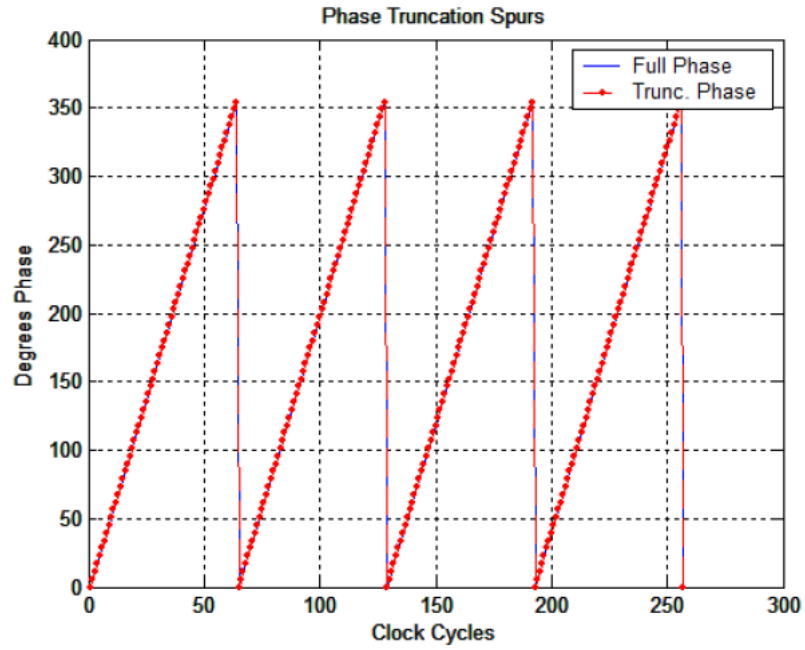


Figura 25: Acumulador de fase não truncado

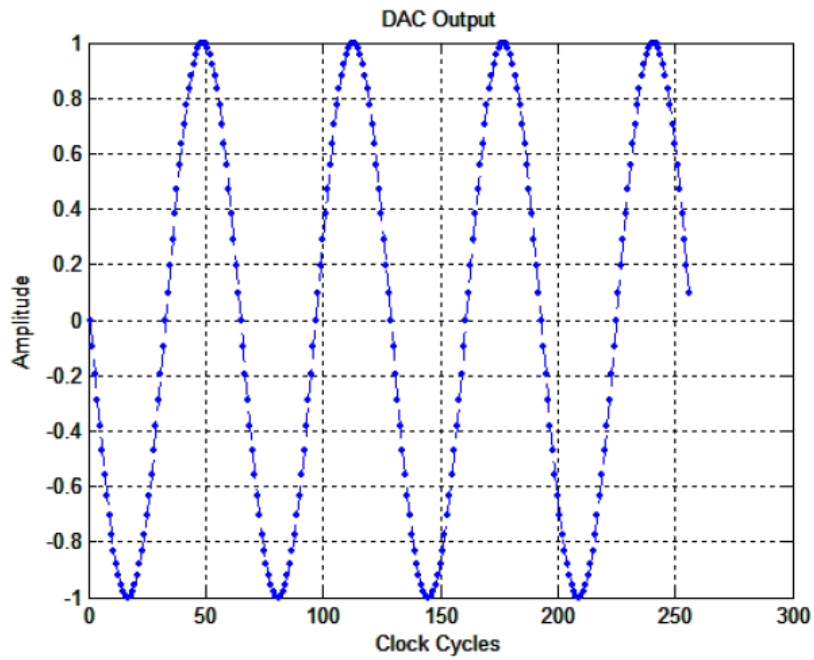


Figura 26: Saída do conversor digital-analógico não truncada

Nas Figuras 27 e 28 [17] foi realizado o truncamento da *LUT* em 3 bits, ou seja, a quantidade de bits após o truncamento foi reduzida de 3 bits. Observando os gráficos notamos o efeito do truncamento de fase, os quais são representados pelos “degraus” vermelhos. As configurações dessas formas de onda são ($N=8$; $L=5$; $M=1$).

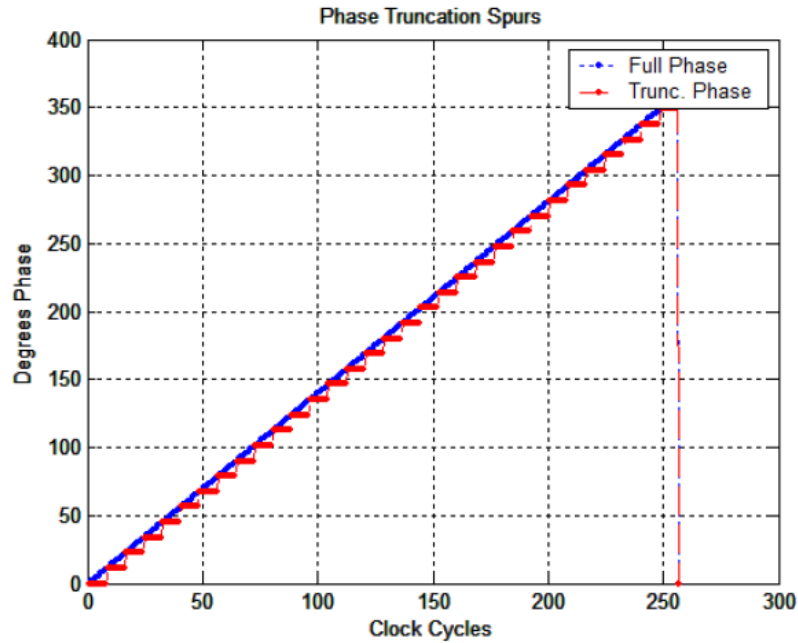


Figura 27: Saída do acumulador truncada.

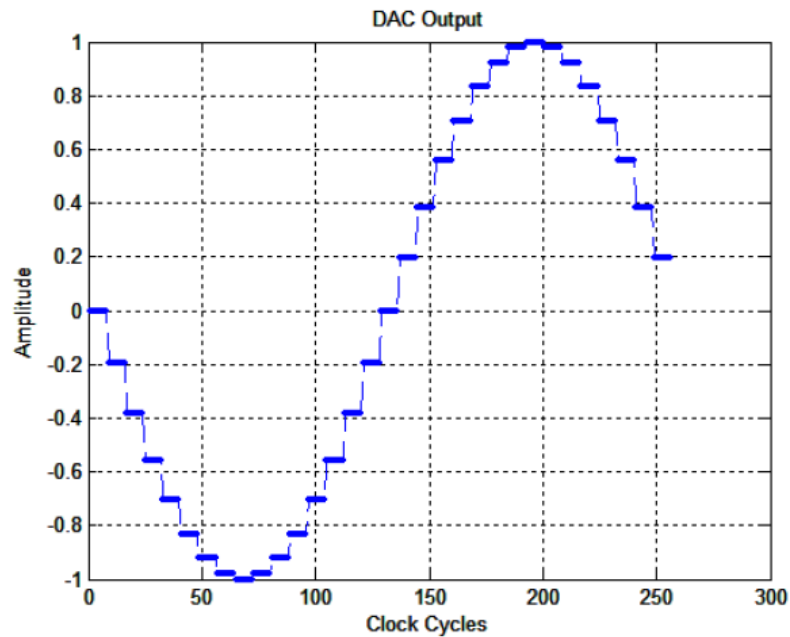


Figura 28: Saída do conversor digital-analógico truncada.

Obviamente, os erros de fase introduzidos pelo truncamento do acumulador irão resultar em erros de amplitude durante o processo de fase para amplitude inerente ao sistema DDS. Contudo esses erros são periódicos. Eles são periódicos porque, independentemente da palavra de ajuste escolhida, após um número suficiente de rotações da roda de fase, o acumulador e o truncamento de fase vão coincidir. Uma vez que estes erros de amplitude são periódicos no domínio do tempo, eles aparecem como linha de espectros (espúrios), no domínio da frequência e são conhecidos como espúrios do truncamento de fase.

Contudo a magnitude e a distribuição dos espúrios do truncamento de fase são dependente três fatores:

1. Tamanho do acumulador (N bits)
2. Tamanho da palavra de fase (P bits), isto é, o número de bits de fase após o truncamento
3. Palavra de ajuste (M)

5.1.1. Magnitude espúria do truncamento de fase

Até o presente momento utilizamos a variável N para nos referir ao tamanho do acumulador ou o tamanho da palavra de ajuste, porém, especificamente neste capítulo será referenciada através da variável A . A maioria do material disponível na literatura, e principal fonte de referência nessa parte da pesquisa, sobre magnitude espúria no truncamento de fase pertence à *Analog Devices*®. Essa por sua vez utiliza a variável A . Caso o leitor venha a pesquisar mais sobre o assunto provavelmente encontrará as principais informações em materiais da *Analog Devices*®.

Com o propósito de evitar confusão e facilitar uma pesquisa futura em materiais da *Analog Devices*® (que é um dos grandes fabricantes do sistema de *DDS*), iremos utilizar, a

partir desse momento, apenas neste capítulo, faremos a troca de variável de N para A para nos referirmos ao tamanho da palavra do acumulador e assim normalizar a nomenclatura.

Certas palavras de ajuste não produzem sinais espúrios de truncamento de fase, contudo outras produzem sinais espúrios de máxima amplitude.

Se a quantidade $A-P$ é de 4 ou mais (geralmente o caso de qualquer arquitetura *DDS*), então o nível máximo de espúrios acaba sendo muito próximo de $-6,02P$ dBc. A unidade dBc equivale ao dB relativo a portadora, no exemplo em questão temos $6,02P$ decibéis abaixo do nível da palavra de ajuste.

Assim, um *DDS* com um acumulador de 32 bits e 12 bits de palavra de fase irá produzir espúrios de truncamento de fase de não mais do que -72 dBc independentemente da palavra de ajuste escolhida [3].

As palavras de ajuste que produzem o valor máximo de espúrios podem ser representadas por:

$$MDC(M, 2^{(A-P)}) = 2^{(A-P-1)} \quad [17]$$

Relembrando o capítulo anterior (para a mudança de N para A):

M : Palavra de ajuste

A : Tamanho da palavra de ajuste (tamanho do acumulador)

P : Número de bits após o truncamento de fase

Onde $MDC(x, y)$ é o máximo divisor comum de x e de y . De fato, para essa equação ser verdadeira, uma palavra de ajuste padrão para a palavra de ajuste é mostrada na Figura 29 [3]:

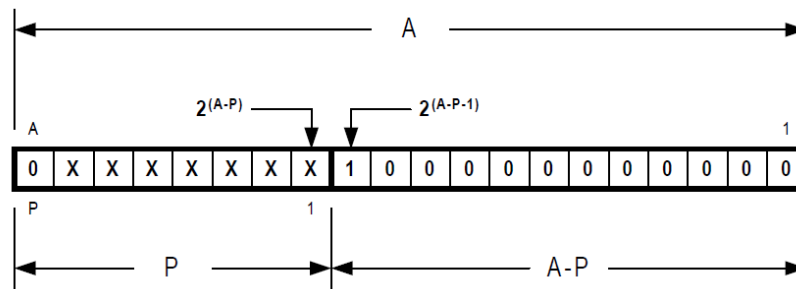


Figura 29: Palavra de ajuste padrão correspondente a máxima magnitude de espúrios.

Uma palavra de A bits é mostrada, o que corresponde a um acumulador de fase de A bits de resolução. Os P bits mais significativos constituem a “palavra de fase” (os bits que são utilizados para a conversão de fase para amplitude). Os $A-P$ bits menos significativos são truncados, isto é, são ignorados na resolução de fase em questão. A palavra de ajuste T , é composta dos $A-1$ bits menos significativos (o bit mais significativo da palavra de ajuste deve ser 0 para evitar o problema de aliasing). Como mostrado na Figura 29, qualquer palavra de ajuste com um 1 na posição do bit $2^{(A-P-1)}$ e 0’s em qualquer posição de bits menos significativos resultam no pior caso da magnitude espúria de fase (-6,02p dBc) [3].

No outro extremo há palavras de ajuste que não produzem espúrias no truncamento de fase, como as palavras de ajuste que satisfazem a relação:

$$MDC(T, 2^{(A-P)}) = 2^{(A-P)} \quad [18]$$

Para que esta equação seja verdadeira, a palavra de ajuste padrão deve ser como mostrada na Figura 30 [3]:

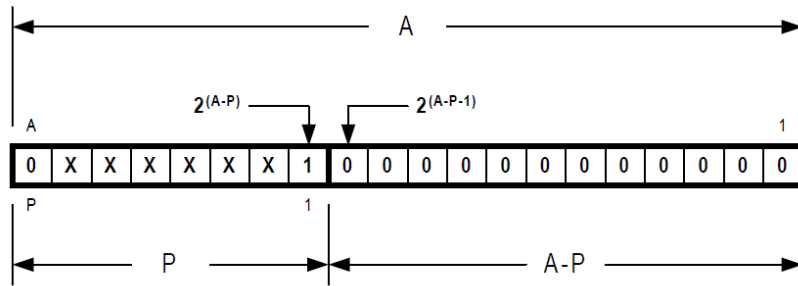


Figura 30: Palavra de ajuste padrão que não resulta em erro espúrios.

Assim, as palavras de ajuste que não produzem espúrios no truncamento de fase são caracterizadas por um 1 na posição do bit $2^{(A-P)}$ e 0's em todas as posições de bits menos significativos. Obviamente, todos os outros padrões de palavras de ajuste os quais não se enquadram nas duas categorias acima irão produzir espúrios de truncamento de fase entre os dois extremos [3].

O acumulador de fase atualiza sua saída após a mudança de estado do relógio. Quando a entrada de controle M não for um divisor inteiro de 2^N , tem-se a geração de um erro no sinal de saída devido à variação da quantidade de amostras solicitadas a LUT . No exemplo acima, nota-se que para $M = 2$, divisor inteiro de 2^N , o sinal na saída do acumulador de fase será periódico. Porém, para $M = 6$, número não múltiplo de 2^N , o sinal de saída gerado é um sinal pseudo-periódico. Ele possui uma quebra de periodicidade que origina a formação do *Jitter* (capítulo 5.5). Para este caso haverá variação do período de leitura da memória durante a geração do sinal, como mostra a Figura 31. Com o valor de $M = 2$, tem-se um ciclo periódico de oito leituras de endereço da LUT a cada varredura de toda a memória. Para $M = 6$ observam-se dois ciclos de leitura da memória, um ciclo com três leituras e outro com duas leituras, o que gera um *Jitter* no sinal de saída.

Observa-se certa periodicidade do erro inserido no sinal de saída e esta pode ser calculada através de [2]:

$$P_e = T \cdot MDC(M, 2^N) \quad [19]$$

Sendo P_e a periodicidade do erro e MDC o máximo divisor comum. Para os casos onde o $MDC(M, 2^N) = 2^N$, não se observa erro gerado na saída. Caso contrário, o erro estará presente com uma periodicidade fixa. O exemplo anterior pode ser útil neste caso.

$$P_e = T \cdot MDC(6,16) = 48 \cdot T = \frac{48}{16} = 3 \text{ ciclos} \quad [20]$$

Que indica que a quebra na periodicidade da leitura da memória acontecerá a cada três ciclos completos de leitura, como pode ser visto na Figura 31:

Como o erro gerado é periódico no domínio tempo, existirá um erro também periódico gerado no domínio da frequência. Este erro será na forma de sinais espúrios presentes no espectro do sinal sintetizado de saída. A periodicidade do erro pode ser calculada pela equação 20. Através de uma análise mais detalhada deste fenômeno, é possível afirmar também que o *Jitter* gerado no sinal de saída será proporcional a relação $\frac{M}{2^N}$. O *Jitter* é um fator importante no momento da escolha dos incrementos de fase.

A Figura 31 mostra a relação entre o incremento da fase e o aparecimento do *Jitter*. Dependendo do valor de M escolhido o acumulador de fase irá gerar em sua saída um sinal perfeitamente periódico ou não. Quanto menos periódico for o sinal maior será o *Jitter* produzido.

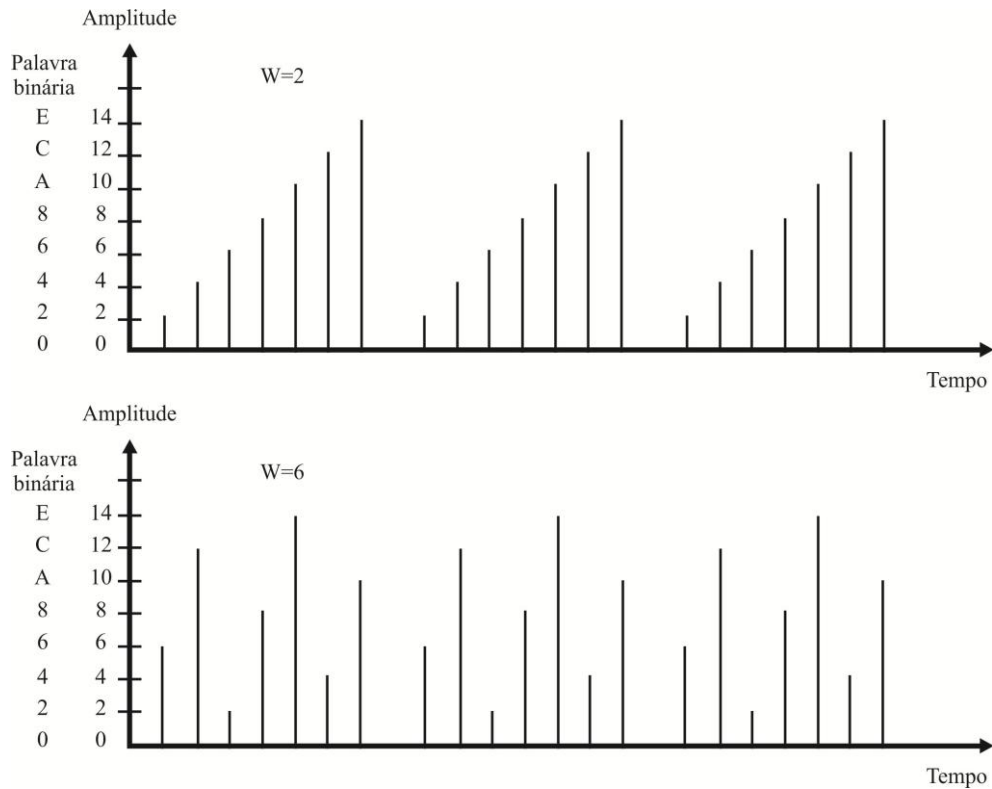


Figura 31: Relação do *Jitter* com o incremento de fase.

5.2. Simetria do Quarto de Onda

Visando otimizar o tamanho da memória *ROM* e minimizar os efeitos do truncamento do número de bits “ruído de quantização”, podem ser implementados vários métodos de compressão da memória. Estes são capazes de diminuir o conteúdo da memória *ROM* sem que haja perdas na qualidade do sinal sintetizado.

A compressão de quarto de onda é baseada nas propriedades de simetria da onda senoidal. Analisando a forma de onda da Figura 32 percebe-se que as informações dos valores do módulo da amplitude da senoide contidas no intervalo de 0° a 90° são as mesmas contidas nos intervalos de 90° a 180° , 180° a 270° e 270° a 360° . Isso significa que se forem armazenadas somente as informações contidas no intervalo de 0° a 90° da onda senoidal na memória *ROM*, um quarto de toda a informação de um ciclo completo da onda senoidal, esta poderá ser totalmente reconstruída sem perda ou deformação [2].

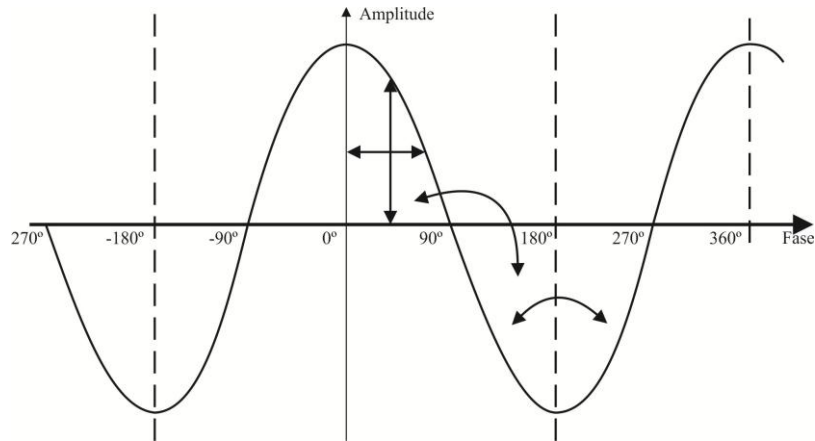


Figura 32: Simetria do quarto de onda.

Após a gravação da memória com informações apenas do primeiro quadrante da senoide, é necessária a construção de um circuito inteligente que manipule os dados armazenados na *ROM*, para reconstruir o sinal de forma correta. Um exemplo deste esquema pode ser visto na Figura 33.

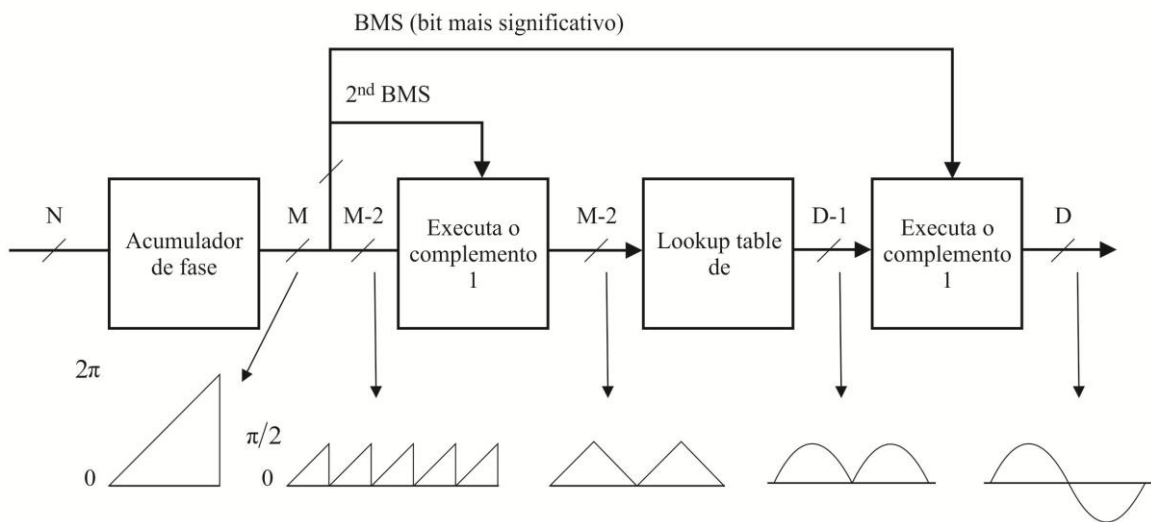


Figura 33: Esquema *DDS* com compressão de quarto de onda.

5.3. Teorema da Amostragem

O teorema da amostragem desenvolvido por Shannon define que qualquer sinal estocástico, com energia finita e largura de faixa limitada, com frequência f_1 amostrado no domínio do tempo só pode ser recuperado sem nenhuma deformação se a frequência de amostragem f_a for maior ou igual a duas vezes o valor da máxima frequência do sinal amostrado $f_a \geq 2 \cdot f_{max}$. Com isto, fica estabelecido que a estrutura *DDS* terá a sua faixa de frequência de saída limitada teoricamente em no máximo 50% da frequência do sinal de amostragem utilizado, chamado de relógio principal, pois para frequências sintetizadas maiores que duas vezes a frequência da amostragem ocorrerá a sobreposição da frequência imagem (réplica) na frequência sintetizada, fenômeno conhecido como “aliasing” que impedirá a recuperação do sinal original. Com essas informações e pelo fato de a forma de onda senoidal ser um sinal determinístico, a síntese de frequência utilizada na estrutura *DDS* é feita por meio de processamentos digitais e analógicos das amostras de uma onda senoidal [2].

Uma compreensão da teoria da amostragem é necessária quando analisamos a saída amostrada de uma solução baseada em *DDS*. O espectro de saída amostrada é ilustrado na Figura 34 [3]. Neste exemplo, a frequência de *clock* f_{clk} é de 300 Mhz e a frequência de saída f_s fundamental é de 80 MHz.

De acordo com o Teorema de Nyquist, a quantidade de amostras por unidade de tempo de um sinal, chamada taxa ou frequência de amostragem, deve ser maior que o dobro da maior frequência contida no sinal a ser amostrado. As imagens que foram criadas pelas amostras no sinal de saída são provenientes de $f_{clk} \pm f_s$. Na da Figura 34 a primeira resposta de imagem, ocorre, no exemplo, em $f_{clk} - f_s$ ou 220 MHz. A 2ª, 3ª, 4ª e 5ª imagens aparecem

em 380, 520, 680 e 820 MHz respectivamente [3]. Note que os valores nulos aparecem nos múltiplos da frequência de amostragem.

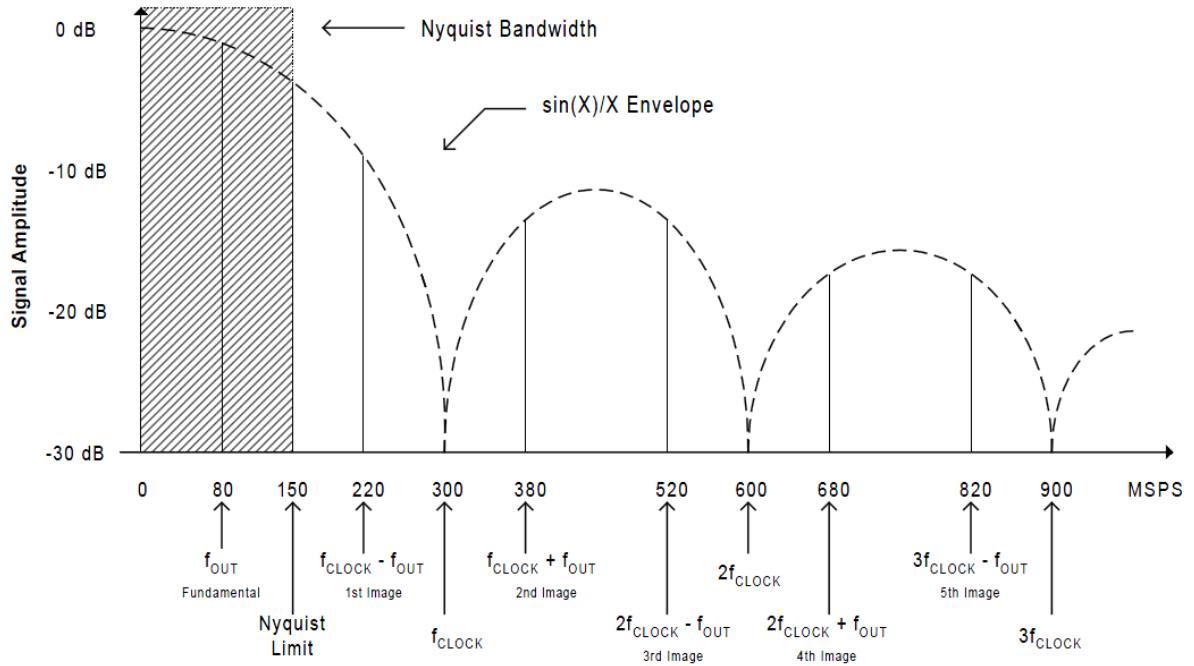


Figura 34: Espectro de amostras do sinal de saída.

Em aplicações típicas de *DDS*, um filtro passa-baixas é utilizado para suprimir os efeitos das respostas imagem no espectro de saída. A fim de manter as restrições da frequência de corte do filtro passa-baixas razoável, é aceito como regra limitar f_s como sendo aproximadamente 40% de f_{clk} . Isto facilita um pouco a implementação de filtros econômicos na saída [20], de menor ordem e menos exigentes em relação à precisão e estabilidade dos componentes.

Como pode ser observado na Figura 34 o espectro é modulado por uma função $sinc = \frac{\sin x}{x}$. Isto é devido a natureza *holding* (manter um dado até o próximo) do conversor digital-analógico. Assim a amplitude da fundamental e qualquer resposta imagem pode ser calculada usando a função *sinc*. Como já foram mencionadas, arquiteturas *DDS* podem incluir um bloco que realiza a compensação aplicando a função inversa de *sinc* [20].

No caso da frequência de saída f_s exceder a frequência f_{clk} , a primeira imagem aparecerá como erro de *aliasing*.

O erro de *aliasing* em processamento digital de sinais e outras disciplinas relacionadas se refere ao efeito de alguns sinais se tornarem indistinguíveis quando são amostrados, o erro de *aliasing* nada mais é que a superposição dos espectros de saída de cada raia da frequência de amostragem, por falta de espaço.

Este problema ocorre pelo fato de que a saída do conversor não é uma série de impulsos de largura zero (como em um perfeito re-amostrador), mas uma série de pulsos retangulares cuja largura é igual ao inverso da taxa de atualização.

Em outras palavras, e de forma prática, qualquer componente de frequência acima $\frac{f_{clk}}{2}$ é indistinguível de uma componente de baixa frequência, chamado de *alias*, associado a uma das cópias.

A partir das considerações acima chegamos a conclusão que:

$$f_{smax} = \frac{f_{clk}}{2} \quad [21]$$

5.4. Quantização

Após a amostragem temos uma seqüência de números os quais podem, teoricamente, ainda assumir qualquer valor em uma faixa contínua de valores. Pelo fato de essa faixa ser contínua, existem infinitos valores possíveis que cada número pode assumir. A fim de se poder representar cada número a partir de uma gama tão grande de valores, seriam necessários infinitos números, algo que não temos. Em vez disso, devemos representar tais números com uma quantidade finita de dígitos, ou seja, após a discretização na variável

tempo, agora temos que discretizar na variável amplitude. A discretização dos valores de amplitude é chamada de quantização [6].

Suponha que a nossa seqüência de valores seja de valores na faixa entre $[-1;+1]$. Agora assuma que devemos representar cada número desse intervalo com apenas duas casas decimais, um antes e outro depois do ponto de interesse. Nossas possíveis amplitudes de valores, serão; $-1; -0,9; \dots; -0,1; 0,0; 0,1; \dots; 0,9; 1,0$. Temos então 21 valores distintos de amplitude para representá-los, denotamos esses números como níveis de quantização Nq .

Cada nível é um passo de 0,1 maior do que o predecessor e denotaremos esse passo de quantização como q . Agora vamos assumir que cada valor da nossa faixa de quantização seja mais próximo da nossa amplitude real: na faixa de $[-0,05;+0,05]$ mapeia o nível de quantização do nível 0 na faixa compreendida entre $[0,05;0,15]$ mapeado com um passo de 0,1 e assim por diante.

Este mapeamento pode ser visualizado na Figura 35 como uma função por partes da nossa variável x . Note que esse mapeamento inclui uma operação de “corte” em ± 1 : valores maiores do que 0,95 são mapeados para o nível 1, não importando quão grandes forem e isso é feito analogamente para valores negativos.

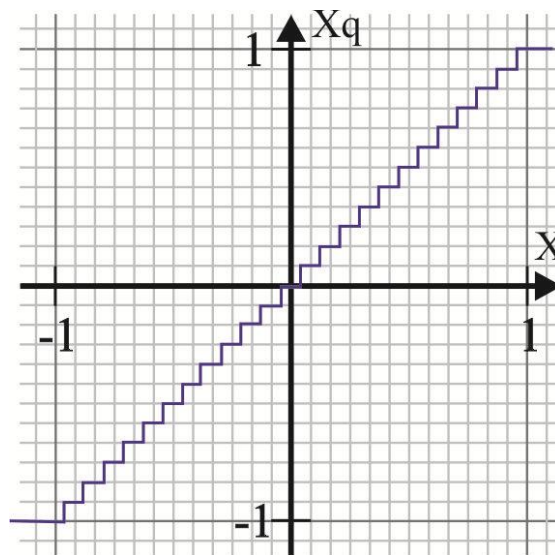


Figura 35: Curva característica da quantização

5.4.1. Efeitos de sobre-amostragem em performance espúria

Lembrando que o princípio básico do teorema de Nyquist define que a quantidade mínima de amostras que devem ser obtidas de um sinal contínuo a ser amostrado deve ser duas vezes a maior frequência deste sinal, a fim de ser possível sua recuperação. Em sobre-amostragem uma taxa de amostragem maior do que o critério de Nyquist é utilizado [10].

A Figura 36 mostra o efeito da sobre-amostragem no *SQR*. O *SQR* ou *Ratio of Signal Power to Quantization Noise Power* é a quantidade de potência de ruído de quantização.

A quantidade da potência de ruído de quantização é dependente da resolução do *DAC*. É uma quantidade fixa e é proporcional a área sombreada. No caso de sobre amostra, a quantidade total da potência de ruído de quantização é o mesmo como no caso de Nyquist amostrado. Uma vez que a potência de ruído é a mesma em ambos os casos (constante), e a área do retângulo que representa o ruído é proporcional à potência de ruído, em seguida, a altura do retângulo de ruído (no caso de sobre amostra) deve ser menor do que o caso de Nyquist amostrados a fim de manter a mesma área [3].

Note-se que na banda de interesse a área do retângulo de ruído é menor para o caso de sobre amostra.

Assim, para uma dada quantidade de potência do sinal na banda de interesse, a relação sinal-ruído é maior quando existe sobre-amostragem.

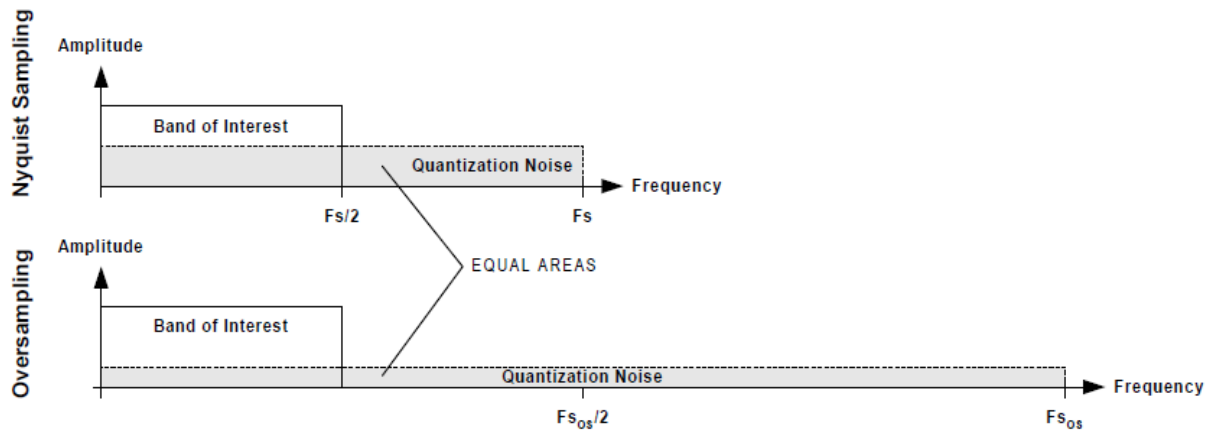


Figura 36: Efeito de sobre-amostragem em SQR

O efeito de sobre-amostragem é quantificável e é dado por:

$$C = 10 \cdot \log \left(\frac{FS_{Os}}{F} \right) [dB] \quad [22]$$

Onde F_s é a taxa de amostragem de Nyquist e FS_{Os} é a taxa de sobre-amostragem.

A equação SQR modificada é a seguinte:

$$\begin{aligned} SQR &= 1,76 + 6,02B + A + C \\ &= 1,76 + 6,02B + 20 \cdot \log(FFS) + 10 \cdot \log \left(\frac{FS_{Os}}{F} \right) [dB] \quad [23] \end{aligned}$$

Fazendo uma comparação com a distorção de quantização, se operarmos com o DAC a 70% da máxima amplitude e com uma sobre-amostra por um fator de 3, o SNR se torna 51,59 dB. Isto constitui uma melhoria global de 1,67 dB.

5.5. Jitter

Jitter é a variação temporal de um conjunto de arestas de sinais a partir de seus valores ideais.

Uma oscilação perfeita teria suas bordas de subida e descida ocorrendo em momentos precisamente iguais e nunca iriam variar. Isto, é claro, é impossível, como até os melhores osciladores são construídos a partir de componentes reais com fontes de ruído e outras imperfeições. A alta qualidade, baixo ruído de fase de cristais osciladores tem *Jitter* com menos de 35 pico segundos (ps), acumulados ao longo de muitos milhões de ciclos de *clock* [4].

Essas instabilidades do *Jitter* podem, eventualmente, provocar deslizamentos ou sinais perdidos que resultam em perda de dados. A Figura 37 mostra uma onda quadrada com *Jitter* comparado para um sinal ideal na mesma frequência.

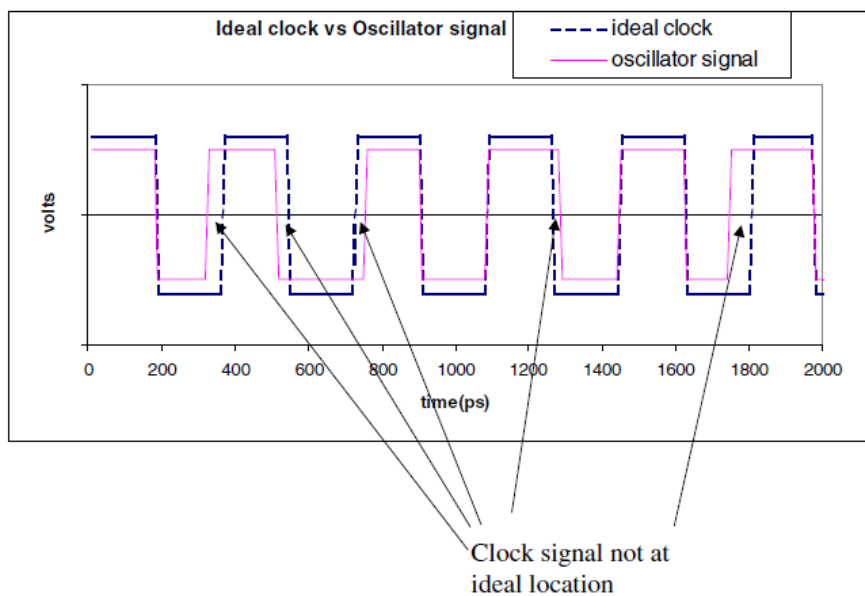


Figura 37: Sinal com instabilidades devidas ao *Jitter*

O resultado é a propagação de uma linha espectral associada a frequência de saída do DDS.

CONCLUSÕES

A capacidade de produzir com precisão e controlar sinais de várias frequências e formas de onda tornou-se um requisito indispensável a um grande número de aplicações modernas. A utilização de arquiteturas digitais se mostrou eficiente por apresentar mais agilidade nas mudanças de frequência e fase em relação aos dispositivos analógicos pois não necessita tempos de acomodação.

O propósito deste trabalho foi avaliar os blocos constituintes método de síntese digital direta, abordando os conceitos básicos sobre sintetizadores de frequência.

A metodologia levou em consideração a descrição detalhada dos blocos operacionais e a formalização matemática das equações que configuraram a frequência do sinal, palavra de ajuste e suas relações com as saídas desejadas.

Por fim com o objetivo de contribuir para que o leitor tivesse subsídios para escolher entre as arquiteturas disponíveis no mercado, uma que fosse suficiente para agregar à aplicação de interesse, foram ainda abordados aspectos mais técnicos em relação à qualidade da onda sintetizada, propagação de variáveis espúrias, não linearidades e truncamento de bits, explorando os limites de operação do método e sugerindo futuros trabalhos.

A implementação de um conjunto de rotinas do *Matlab*® permitiu visualizar os sinais ao longo do *DDS* e avaliar o impacto das decisões de projeto sobre o resultado final.

REFERÊNCIAS

- [1] VIDAL, Félix Fedatto. Explorando o campo da síntese digital direta visando uma aplicação.
- [2] CARDOSO, Rodrigo Otávio Rocha. Geração de sinais de referência para transmissores de televisão.
- [3] A Technical Tutorial on Digital Signal Synthesis, 1999, Analog Devices, Inc.
- [4] Eva Murphy and Colm Slattery, Analog Dialogue 38-08, August (2004).
- [5] IEEE Signal Processing Magazine July 2004, Vol. 21, Num. 4
- [6] Digital Signals – Sampling and Quantization. Disponível em: <www.rs-met.com> Acesso em: 10 abr. 2012
- [7] Understanding *Direct Digital Synthesis (DDS)*. Disponível em: <<http://zone.ni.com/devzone/cda/tut/p/id/5516#toc2>> Acesso em: 10 abr. 2012
- [8] MARTINS, Renato da Costa. Gerador de funções digital sintetizado. 2003. 65 f. Monografia (Bacharelado em Engenharia Elétrica)-Universidade de Brasília, Brasília, 2003.
- [9] Full scale. Disponível em: <http://en.wikipedia.org/wiki/Full_scale> Acesso em: 10 abr. 2012
- [10] Teorema de Nyquist. Disponível em: <<http://pt.scribd.com/doc/51945098/TEOREMA-DE-NYQUIST>> Acesso em: 15 abr. 2012
- [11] Ciciliato, Vitor. Sintetizador de frequências empregando a tecnica de sintese digital direta coerente.
- [12] Apresentação Analog Devices
- [13] MT-085. TUTORIAL. Fundamentals of *Direct Digital Synthesis (DDS)*.
- [14] Vankka, Jouko. Direct Digital Synthesizers: Theory, Design and Applications.
- [15] Pápay, Zs. Numerical Distortion in Single-Tone *DDS*
- [16] Nikolic, Ognjen. Direct Digital Synthesizer (*DDS*)

- [17] Nguyen, Tri Trong. Direct Digital Frequency Synthesizer Architecture for Wireless Communication in 90 Nm CMOS Technology
- [18] Funcionamento do bluetooth. Disponível em:
<<http://informatica.hsw.uol.com.br/bluetooth1.htm>> Acesso em: 1 mai. 2012
- [19] CARDOSO, Adriano dos Santos. Desenvolvimento e Implementação de um Sintetizador de Frequência CMOS utilizando Sistema Digital.
- [20] Tecnologia Bluetooth. Disponível em:<<http://www.infowester.com/bluetooth.php>>
Acesso em: 1 mai. 2012
- [21] Rusterholz, Thomas. Keller, Marcel. Direct Digital Synthesizer (*DDS*) mit komprimierter Sinustabelle

Apêndice

Matlab® - Código 1

```

%-----
% Simulação em Matlab® do método de
% Síntese Digital Direta
%-----
% Rotina 1
% Implementa o comportamento básico dos 3 blocos principais

clear all;          %Limpa a memoria anterior

L = 5;              % 2^L - Enderecos na LUT
N = 8;              % Bits no acumulador de fase
M = 4;              % Magnitude do incremento
i = 0;              % Variável auxiliar
fase = 0;           % Saída de fase

% Configuração das amplitudes da senoide na LUT
% Dividir 360 graus de fase nas 2^L partes

for x = -pi:2*pi/(2^L):pi-(2*pi/(2^L))
    i = i + 1; % De 0 a 31
    LUT(i)=sin(x);
end

t = 0:1:2^L-1;
plot(t, LUT, 'bs');
title('Amplitude senoide LUT');
xlabel('Enderecos');
ylabel('Amplitude');
grid on
pause;

% Acumulador de fase
for x = 2:2^N
    fase (x) = fase (x-1)+M;
    if fase(x)>2^N-1
        fase (x)=fase (x)-2^N;
    end
end

t = 1:1:2^N;
plot(t, fase, '--b. ');
title('Acumulador de fase');
xlabel('Ciclos de clock');
ylabel('Saida de fase');
grid on
pause;

% Decodificador fase/amplitude
for x = 1:2^N
    addr=int32 (fase (x)/2^(N-L)+1);
    if addr<=32
        ampl (x)=LUT (addr);
    end
end
end

```

```

ampl(256) = ampl(255);

pause;
plot(t, ampl, '--b. ');
title('Saida do DAC');
xlabel('Ciclos de clock');
ylabel('Amplitude');
grid on
pause;
%-----

```

Matlab® - Código 2

```

%-----
% Rotina 2
% Implementa o comportamento da saída digital de fase

% Determinação do fator de frequência como divisor do
% clock máximo por 2^N, ou seja, a menor frequência
X = 2^6;

% Número de períodos simulados. Deve ser um número inteiro
SimT=2;

% Resolução de simulação.
% Resolução de um "step" do sinal de saída do DAC
SimR=2^2;

% A FFT será realizada com "N" pontos
N=X*SimR;

% Tempo de simulação
SimD=SimT*N;

% Frequência do clock
ClkF=20E6;

% Número de bits do acumulador
AccuN=32;

% Número de bits que a LUT armazena,
% "barramento de endereços"
LutN=22;

% Número de bits do DAC
DacN=8;

% Determinação da frequência
f = max(ClkF/X, ClkF/2^LutN);
dAccuPhi=2^AccuN*f/ClkF;

% Tempo de simulação como número inteiro de ciclos de clock
n=(0:(SimD-1))/SimR;

% Tempo de simulação discreta
t= n*1/ClkF;

% Fase discreta do tempo de simulação para SimD

```

```

nphi=zeros(1, SimR);
for i=1:(X*SimT-1)
    nphi=[nphi, ones(1, SimR)*i];
end
AccuPhi=rem(nphi*dAccuPhi, 2^AccuN);

% Pega apenas o bit mais significativo
LutPhi=floor(AccuPhi/2^(AccuN-LutN));

% Funções para plotar o gráfico final
plot(t, LutPhi)
grid on
title('Saída Digital de Fase')
xlabel('Tempo [s]')
ylabel('Fase [bits]')
axis([0 SimT/f 0 2^LutN])
%-----

```

Matlab® - Código 3

```

%-----
% Rotina 3
% Implementa o comportamento do sinal de saída do conversor DA
% Serão comentados apenas os itens novos em relação
% a rotina 2

X = 2^5;
SimT=2;
SimR=2^2;
N=X*SimR;
SimD=SimT*N;

ClkF=20E6;
AccuN=32;
LutN=22;
DacN=8;

f = max(ClkF/X, ClkF/2^LutN);
dAccuPhi=2^AccuN*f/ClkF;
n=(0:(SimD-1))/SimR;

t= n*1/ClkF;

nphi=zeros(1, SimR);
for i=1:(X*SimT-1)
    nphi=[nphi, ones(1, SimR)*i];
end
AccuPhi=rem(nphi*dAccuPhi, 2^AccuN);

LutPhi=floor(AccuPhi/2^(AccuN-LutN));

% Valor máximo digital
maxWert=2^(DacN-1)-1;

% Saída dos valores de erro de quantização do DAC
kontY=sin(LutPhi*2*pi/2^LutN);
diskrY=round(kontY*maxWert)/maxWert;

```

```

kontY_DC=mean(kontY);
diskrY_DC=mean(diskrY);
qN_DC=mean(diskrY-kontY);
qN=diskrY-kontY-qN_DC;

```

```

plot(t, diskrY, 'g')
grid on
title('Saída do Conversor')
xlabel('Tempo [s]')
ylabel('Tensão DC [1]')
axis([0 SimT/f -1 1])
%-----

```

Matlab® - Código 4

```

%-----
% Rotina 4
% Implementa o espectro de saída do sinal do conversor DA
% Serão comentados apenas os itens novos em relação
% às rotinas anteriores

X = 2^8;
SimT=8;
SimR=2^6;
N=X*SimR;
SimD=SimT*N;

ClkF=20E6;
AccuN=32;
LutN=22;
DacN=8;

f = max(ClkF/X, ClkF/2^LutN);
dAccuPhi=2^AccuN*f/ClkF;

n=(0:(SimD-1))/SimR;
t= n*1/ClkF;
nphi=zeros(1, SimR);
for i=1:(X*SimT-1)
    nphi=[nphi, ones(1, SimR)*i];
end
AccuPhi=rem(nphi*dAccuPhi, 2^AccuN);
LutPhi=floor(AccuPhi/2^(AccuN-LutN));
maxWert=2^(DacN-1)-1;
kontY=sin(LutPhi*2*pi/2^LutN);
diskrY=round(kontY*maxWert)/maxWert;

% Espectro de Y
kontYspecA=2*abs(fft(kontY))./SimD;
diskrYspecA=2*abs(fft(diskrY))./SimD;

% Espectro de frequência de Y
specF=n*f*(SimR/SimT);

qN=diskrY-kontY-mean(diskrY-kontY);

% Potência do sinal em nível de ruído
SNR=10*log10(sum(kontY.^2)/sum(qN.^2));

```

```

% Ganho na relação sinal/ruído
SNRgewinn=10*log10(SimD)-3;
SimNoiseFloor=SNR+SNRgewinn;

bar(specF, 150 + 20*log10(diskrYspecA/max(diskrYspecA)));
grid on
title('Espectro de Saída do Conversor')
xlabel('Frequencia [Hz]')
ylabel('Amplitude [dB] normalizada para o maximo + 150 dB')
axis([0 SimR*ClkF/2 40 150])

maximum=20*log10(max(diskrYspecA))
%-----

```

Matlab® - Código 5

```

%-----
% Rotina 5
% Implementa o ruído de Quantização de saída do sinal do conversor DA
% Serão comentados apenas os itens novos em relação
% às rotinas anteriores

X = 2^8;
SimT=1;
SimR=2^8;
N=X*SimR;
SimD=SimT*N;

ClkF=20E6;
AccuN=32;
LutN=22;
DacN=8;

f = max(ClkF/X, ClkF/2^LutN);
dAccuPhi=2^AccuN*f/ClkF;

n=(0:(SimD-1))/SimR;
t= n*1/ClkF;
nphi=zeros(1, SimR);
for i=1:(X*SimT-1)
    nphi=[nphi, ones(1, SimR)*i];
end

AccuPhi=rem(nphi*dAccuPhi, 2^AccuN);
LutPhi=floor(AccuPhi/2^(AccuN-LutN));
maxWert=2^(DacN-1)-1;
kontY=sin(LutPhi*2*pi/2^LutN);
diskrY=round(kontY*maxWert)/maxWert;

kontY_DC=mean(kontY);
diskrY_DC=mean(diskrY);
qN_DC=mean(diskrY-kontY);
qN=diskrY-kontY-qN_DC;

plot(t, qN)
grid on
title('Quantização do sinal de saída do DAC')
xlabel('Tempo [s]')
ylabel('Tensão AC [1]')

```

```
axis([0 (SimT/f)/7 -4e-3 4e-3])
```

```
%-----
```

Matlab® - Código 6

```
%-----
```

```
% Rotina 6
```

```
% Implementa o sinal de saída do conversor após filtro de ordem 2
```

```
% Serão comentados apenas os itens novos em relação
```

```
% às rotinas anteriores
```

```
X = 2^6;
```

```
SimT=2;
```

```
SimR=2^4;
```

```
N=X*SimR;
```

```
SimD=SimT*N;
```

```
ClkF=20E6;
```

```
AccuN=32;
```

```
LutN=22;
```

```
DacN=8;
```

```
f = max(ClkF/X, ClkF/2^LutN);
```

```
dAccuPhi=2^AccuN*f/ClkF;
```

```
n=(0:(SimD-1))/SimR;
```

```
t= n*1/ClkF;
```

```
nphi=zeros(1, SimR);
```

```
for i=1:(X*SimT-1)
```

```
    nphi=[nphi, ones(1, SimR)*i];
```

```
end
```

```
AccuPhi=rem(nphi*dAccuPhi, 2^AccuN);
```

```
LutPhi=floor(AccuPhi/2^(AccuN-LutN));
```

```
maxWert=2^(DacN-1)-1;
```

```
kontY=sin(LutPhi*2*pi/2^LutN);
```

```
diskrY=round(kontY*maxWert)/maxWert;
```

```
kontYspecA=2*abs(fft(kontY))./SimD;
```

```
diskrYspecA=2*abs(fft(diskrY))./SimD;
```

```
specF=n*f*(SimR/SimT);
```

```
fg=ClkF/2;
```

```
tp2=(1+j*specF/fg).^(-2);
```

```
tp=diskrYspecA.*tp2;
```

```
tps=abs(fft(tp)).*sin(angle(fft(tp)))*SimD;
```

```
mean(tps)
```

```
plot(t, tps, 'g')
```

```
grid on
```

```
title('Saída do sinal do DAC com filtro de ordem 2')
```

```
xlabel('Tempo [s]')
```

```
ylabel('Tensão DC [1]')
```

```
axis([0 SimT/f -1 1])
```

```
%-----
```